

**Titre:** Aide à la prise de décision en temps réel dans un contexte de  
Title: production adaptative

**Auteur:** Benoît Saenz de Ugarte  
Author:

**Date:** 2009

**Type:** Mémoire ou thèse / Dissertation or Thesis

**Référence:** Saenz de Ugarte, B. (2009). Aide à la prise de décision en temps réel dans un  
Citation: contexte de production adaptative [Ph.D. thesis, École Polytechnique de  
Montréal]. PolyPublie. <https://publications.polymtl.ca/164/>

 **Document en libre accès dans PolyPublie**  
Open Access document in PolyPublie

**URL de PolyPublie:** <https://publications.polymtl.ca/164/>  
PolyPublie URL:

**Directeurs de  
recherche:** Robert Pellerin, & Abdelhakim Artiba  
Advisors:

**Programme:** Doctorat en génie industriel  
Program:

UNIVERSITÉ DE MONTRÉAL

**AIDE À LA PRISE DE DÉCISION EN TEMPS RÉEL  
DANS UN CONTEXTE DE PRODUCTION ADAPTATIVE**

BENOÎT SAENZ DE UGARTE

DÉPARTEMENT DE MATHÉMATIQUES ET DE GÉNIE INDUSTRIEL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

THÈSE PRÉSENTÉE EN VUE DE L'OBTENTION

DU DIPLÔME DE PHILOSOPHIÆ DOCTOR

(GÉNIE INDUSTRIEL)

NOVEMBRE 2009

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Cette thèse intitulée :

AIDE À LA PRISE DE DÉCISION EN TEMPS RÉEL DANS UN CONTEXTE DE  
PRODUCTION ADAPTATIVE

présentée par : SAENZ DE UGARTE Benoît

en vue de l'obtention du diplôme de : Philosophiæ Doctor

a été dûment acceptée par le jury d'examen constitué de :

M. FRAYRET Jean-Marc, Ph.D., président

M. PELLERIN Robert, ing., Ph.D., membre et directeur de recherche

M. ARTIBA Abdelhakim, Docteur, membre et codirecteur de recherche

M. TRÉPANIÉ Martin, ing., Ph.D., membre

M. LAMOURI Samir, Docteur, membre externe

## REMERCIEMENTS

Je tiens à remercier chaleureusement mon directeur de recherche, M. Robert Pellerin, professeur à l'École Polytechnique de Montréal, et mon codirecteur de recherche, M. Abdelhakim Artiba, professeur à l'université de Valenciennes et du Hainaut-Cambrésis, qui ont rendu possible ce travail par leur écoute, leur soutien et leurs judicieux conseils.

Je pense ici en particulier à M. Jean-Marc Frayret, qui m'a fait l'honneur de présider le Jury de cette thèse et à MM. Martin Trépanier et Samir Lamouri qui ont accepté d'être membre du Jury; je les en remercie profondément.

Mes remerciements vont également à M. Charles-André Horth de STI Corp pour nous avoir proposé des scénarios industriels réalistes, pour nous avoir fourni suffisamment d'information pour mener à bien notre expérimentation et pour m'avoir permis d'effectuer deux stages très enrichissants dans son entreprise. Je tiens aussi à mentionner le plaisir que j'ai eu à travailler au sein de STI Corp, et j'en remercie ici tous les membres.

J'exprime aussi ma gratitude au CRSNG pour le financement de cette thèse et au CIRRELT pour l'octroi d'une bourse d'excellence.

Je remercie M. Adnène Hajji, pour avoir bien voulu partager ses connaissances et ses avisés conseils en simulation. Je remercie également Loïc, Michel, Jean-François, Coralie, Hélène, Houaida, Sandrine et Jean-Baptiste du laboratoire de recherche sur les processus manufacturiers adaptatifs de l'École Polytechnique pour la bonne humeur qu'ils ont apporté au quotidien.

J'ai une pensée toute particulière pour ma famille. Je remercie mes parents pour leur aide et leur appui moral malgré la distance qui nous sépare.

Enfin, un remerciement tout particulier à ma fiancée, M<sup>elle</sup> Marcela Santos do Carmo pour avoir su apaiser mes doutes et pour m'avoir apporté son indéfectible support, sa confiance sans faille, sa patience sans borne et sa ténacité.

## RÉSUMÉ

La dynamique des marchés a évolué et les entreprises manufacturières ont dû s'adapter pour rester compétitives. Une usine se définissait historiquement par les biens qu'elle produisait. La valeur de ces biens était évaluée avant tout par leurs composants. Mais sous la pression du marché et de son dynamisme accru, les usines souhaitant rester compétitives deviennent de plus en plus des centres de service. Cela provoque des changements et des problèmes de gestion pour lesquels elles n'étaient pas préparées. L'efficacité économique de la création de valeur n'est plus aujourd'hui la seule propriété des produits, mais se déplace vers les processus. Cela signifie que les potentiels qui seront décisifs ne sont pas à chercher dans les capacités des produits, mais dans les capacités des processus. En effet, la mondialisation accroît l'anonymat des produits tout au long de chaînes d'approvisionnement plus longues et plus complexes. Toute entreprise souhaitant se démarquer de la compétition doit proposer à ses clients de la valeur ajoutée additionnelle tels qu'une flexibilité accrue, des délais de livraison plus courts, un meilleur respect des délais, un plus grand choix d'options. Ces propriétés sont le fruit des processus. Leur valeur ajoutée se transfère à leur résultat et donc au client. Une des conditions nécessaires à la transparence des processus est leur capacité à coller en temps réel au flux de valeur de l'entreprise. Les processus doivent être en mesure de s'adapter aux conditions changeantes de l'environnement, de réagir à des événements imprévus et de résoudre ces difficultés en collaborant. C'est à ces conditions qu'ils pourront devenir des processus adaptatifs.

Cette thèse s'intéresse aux processus de réordonnancement en milieu industriel. Elle vise l'implantation de composantes d'aide à la prise de décision en temps réel ainsi que des mécanismes de boucle rétroactive intégrant l'optimisation et les techniques de simulation au sein d'applications ERP et MES permettant ainsi de connecter l'atelier de production au reste de l'entreprise. La plateforme qui a été mise en place permet de répondre en temps réel aux divers aléas survenant dans l'atelier et peut être étendue au-delà de la problématique de l'ordonnancement.

Les travaux réalisés dans le cadre de cette thèse sont articulés autour de quatre articles publiés, acceptés ou soumis à des revues scientifiques :

- « *Manufacturing execution system – a literature review* », publié dans *Production Planning & Control*, Taylor & Francis. Cet article fait une revue de la littérature des MES et montre que le support à la prise de décision en temps réel fait toujours défaut au traditionnel couple ERP-MES.
- « *Engineering Change Order Processing in ERP Systems: An Integrated Reactive Model* », accepté par *European Journal of Industrial Engineering*, Inderscience Publisher. Cet article propose une première plateforme d'exécution en temps réel intégrant un ERP et un MES à un noyau d'aide à la prise de décision combinant un algorithme génétique et un simulateur. La plateforme ainsi constituée est évaluée sur un cas d'étude de traitement d'un ordre de changement technique dans l'industrie aéronautique.
- « *Development and integration of a reactive real-time decision support system in the aluminum industry* », publié dans *Engineering Applications of Artificial Intelligence*, Elsevier. Cet article propose une version plus flexible et robuste de la plateforme d'exécution en temps réel en y ajoutant un serveur d'intégration. La plateforme est évaluée sur un cas d'étude d'ordonnancement de la production d'alliage dans l'industrie de l'aluminium.
- « *An Improved Genetic Algorithm Approach for On-line Optimization Problems* », soumis à *Production Planning & Control*, Taylor & Francis. Cet article se place dans le contexte d'optimisation en réel où la procédure d'évaluation des solutions générée par un algorithme génétique est longue. Deux améliorations sont proposées, une mémoire cache et un mécanisme de prédation pour relaxer l'évaluation. Ces mécanismes permettent de réduire à la fois le temps de calcul et d'améliorer la qualité de la solution retournée par l'algorithme génétique.

## ABSTRACT

The market dynamics have evolved and manufacturing facilities have followed this trend to stay competitive. The classic factory has been defined by its manufactured goods. The value of these goods has been measured primarily by their material components. But under the market pressure and its increasing dynamism, factories wishing to stay competitive are becoming modern service centers. It has resulted in management problems for which many companies are not yet prepared. Today, the economic efficiency of value creation is not a property of the products but rather of the process. It means the decisive potentials of companies are to be found not so much in their production capability but in their process capability. Indeed, increasing globalization is necessarily leading towards more anonymous products out of long supply chains. Any enterprise wishing to stand out from the competition in the future needs a strategy which offers the customer an additional added value, such as, for example, high flexibility, short delivery times, high delivery reliability, and wide range of variants. These properties are created by the processes. The requirement for process capability gives rise in turn to the requirement that all value-adding processes be geared to the process result and thus to the customer. A necessary condition of process transparency is the ability to map the company's value stream in real time. Processes must be able to adapt to environmental changing conditions, react to unforeseen events and to solve these difficulties by collaborating. Under these conditions they can be called adaptive processes.

This thesis focuses on scheduling process in manufacturing environments. The main objective is to implement real time decision-making support components as well as feedback loop mechanisms integrating optimization and simulation techniques in ERP and MES applications allowing connecting the shop floor to the rest of the enterprise. The proposed platform responds in real time to various events occurring on the shop floor and may be extended beyond the scheduling issue.

The works developed during this thesis are based on four published, accepted or submitted papers to specialized papers:

- “*Manufacturing execution system – a literature review*”, published in *Production Planning & Control*, Taylor & Francis. This paper is a MES literature review and demonstrates that decision-making support is still missing in traditional ERP-MES couple.
- “*Engineering Change Order Processing in ERP Systems: An Integrated Reactive Model*”, accepted by *European Journal of Industrial Engineering*, Inderscience Publisher. This paper proposes a generic manufacturing execution platform integrating ERP and MES to a real-time decision-making nucleus combining a genetic algorithm and a simulator. An Engineering Change Order processing scenario, based on a real scenario in the aerospace industry, is used to validate the proposed platform is validated applied on a case study of a given order of technical change in the aviation industry.
- “*Development and integration of a reactive real-time decision support system in the aluminum industry*”, published in *Engineering Applications of Artificial Intelligence*, Elsevier. This paper proposes a more flexible and robust of our manufacturing execution platform by adding an integration server. The platform is evaluated on a case study of alloy production scheduling in the aluminum industry.
- “*An Improved Genetic Algorithm Approach for On-line Optimization Problems*”, submitted to *Production Planning & Control*, Taylor & Francis. This paper aims at enhancing standard genetic algorithm to reduce the execution time and improve the solution quality when the objective function is expensive to evaluate. More precisely, we propose to add a cache memory to avoid re-testing an already tested solution and a predator mechanism to relax evaluation. These mechanisms reduce the run time and improve the solution quality.



# TABLE DES MATIÈRES

REMERCIEMENTS .....	iii
RÉSUMÉ.....	iv
ABSTRACT .....	vi
TABLE DES MATIÈRES .....	viii
LISTE DES TABLEAUX.....	x
LISTE DES FIGURES.....	xi
LISTE DES SIGLES ET ABRÉVIATIONS .....	xiii
LISTE DES ANNEXES.....	xiv
INTRODUCTION.....	1
CHAPITRE 1: REVUE DE LA LITTÉRATURE.....	6
1.1 Production adaptative.....	6
1.1.1 Définitions.....	7
1.1.1.1 Agilité.....	7
1.1.1.2 Production adaptative.....	10
1.1.2 Défis industriels et difficultés .....	14
1.1.2.1 Hétérogénéité du paysage industriel .....	15
1.1.2.2 Processus décisionnel.....	18
1.2 Stratégie de production adaptative .....	21
1.2.1 Détecter .....	21
1.2.1.1 Connectivité et réseau .....	22
1.2.1.2 Architectures et modèles .....	25
1.2.2 Comprendre.....	27
1.2.3 Décider – Modèles d’optimisation en temps réel.....	28
1.2.3.1 Modèles basés sur la simulation.....	29
1.2.3.2 Modèles basés sur un système expert.....	31
1.2.3.3 Modèles basés sur la logique floue et les réseaux de neurones.....	32

1.2.3.4	Modèles basés sur les réseaux de Petri.....	33
1.2.4	Agir – Propagation de la décision .....	33
1.3	Conclusion.....	36
CHAPITRE 2: SYNTHÈSE .....		37
2.1	Problématique.....	37
2.2	Démarche et objectifs de recherche .....	40
2.3	Article 1 – Revue de la littérature sur les MES .....	42
2.4	Articles 2 et 3 – une plateforme d’exécution en temps réel réactive et intégrée..	44
2.4.1	Application démonstrative dans l’industrie aéronautique .....	45
2.4.2	Application démonstrative dans l’industrie de l’aluminium .....	46
2.5	Article 4 – Mémoire et mécanisme de prédation .....	49
2.6	Conclusion.....	52
CHAPITRE 3: DISCUSSION GÉNÉRALE .....		53
3.1	Système manufacturier adaptatif intelligent.....	53
3.2	Stratégie adaptative .....	54
3.3	Optimisation et faisabilité .....	54
3.4	Place du planificateur dans le processus de décision .....	55
3.5	Qualité d’un simulateur utilisé en ligne .....	56
3.6	Opportunités de recherche.....	58
3.7	Conclusion.....	59
CONCLUSION .....		60
BIBLIOGRAPHIE .....		64
ANNEXES .....		76

## LISTE DES TABLEAUX

Tableau 1.1 – Perturbations, leur source et les systèmes impliqués dans leur détection .....	23
Table B.1 – Processing time and routings.....	121
Table B.2 – Tooling availability periods.....	122
Table B.3 – ECO Processing time and routing .....	127
Table B.4 – Tooling availability periods for the 2nd run.....	129
Table B.5 – ECO Processing time and routing .....	129
Table B.6 – Obtained results for the 2nd run .....	129
Table B.7 – Tooling availability periods for the 3rd run .....	130
Table B.8 – Obtained results for the 3d run .....	130
Table B.9 – Comparative study .....	132
Table C.1 – Quantity and product details.....	149
Table C.2 – Penalty function example with three qualities of aluminum and alloy .....	151
Table C.3 – Reference sequence results.....	157
Table C.4 – Reference sequence results (GA) .....	158
Table C.5 – Comparative study .....	159
Table D.1 – Problem parameters.....	173
Table D.2 – Simple Genetic Algorithm with and without cache memory .....	180
Table D.3 – Predator and Genetic Algorithm – Static tolerance.....	181
Table D.4 – Predator and Genetic Algorithm – Static vs. dynamic tolerance .....	183

## LISTE DES FIGURES

Figure 1.1 – Décomposition d’une stratégie de production adaptative.....	12
Figure 1.2 – Production adaptative vs. Agilité.....	13
Figure 1.3 – Approches de contrôle en temps réel.....	29
Figure 1.4 – Méthodes d’optimisation basées sur la simulation .....	30
Figure 2.1 – Système manufacturier adaptatif intelligent .....	38
Figure A.1 – environment and its layers .....	78
Figure A.2 – MES functionalities .....	81
Figure A.3 – MES positioning in the Computer Integrated Manufacturing (CIM) context	82
Figure A.4 – Adaptive Intelligent Manufacturing System.....	97
Figure B.1 – Real time execution architecture.....	117
Figure B.2 – ECO scenario system workflow.....	119
Figure B.3 – ECO tested scenarios.....	120
Figure B.4 – Simulation / GA optimization module .....	123
Figure B.5 – Simulation model bloc diagram .....	125
Figure B.6 – Obtained results.....	128
Figure B.7 – Execution time evolution .....	133
Figure B.8 – Makespan evolution .....	133
Figure C.1 – System environment.....	144
Figure C.2 – Real time execution architecture.....	146
Figure C.3 – Simulation / GA optimization module logic .....	147
Figure C.4 – Simulation model .....	150
Figure C.5 – Downgrade .....	151
Figure C.6 – Simulation models and interactions .....	152
Figure C.7 – Scenario system workflow .....	156
Figure C.8 – Messages processing time .....	160
Figure D.1 – Simple Genetic Algorithm with memory.....	169

Figure D.2 – Genetic Algorithm with inter-generational memory.....	170
Figure D.3 – Predator evaluation process .....	171
Figure D.4 – Real time execution architecture.....	176
Figure D.5 – Permutation coding .....	178
Figure D.6 – Crossover .....	179
Figure D.7 – Mutation .....	179
Figure D.8 – Predator and Simple Genetic Algorithm.....	182

## **LISTE DES SIGLES ET ABRÉVIATIONS**

APS	Advanced Planning System
B2MML	Business To Manufacturing Markup Language
CIM	Computer-integrated Manufacturing
CSV	Comma-Separated Values
DOM	Document Object Model
ECO	Engineering Change Order
ERP	Enterprise Resource Planning
GPU	Graphical Processor Unit
MES	Manufacturing Execution System
MESA	MES Association
MRPII	Manufacturing Resource Planning
ISA	Instrumentation, Systems & Automation
MRP	Material Requirements Planning
OLTP	On-Line Transactional Processing
OLAP	On-Line Analytical Processing
OPC	Open Process Control ou OLE for Process Control
PLM	Product Lifecycle Management
PRM	Purdue University Reference Model
SCM	Supply Chain Management
SOA	Service-Oriented Architecture
TQM	Total Quality Management

## **LISTE DES ANNEXES**

ANNEXE A: MANUFACTURING EXECUTION SYSTEM – A LITERATURE REVIEW .....	76
ANNEXE B: ENGINEERING CHANGE ORDER PROCESSING IN ERP SYSTEMS: AN INTEGRATED REACTIVE MODEL .....	110
ANNEXE C: DEVELOPMENT AND INTEGRATION OF A REACTIVE REAL-TIME DECISION SUPPORT SYSTEM IN THE ALUMINUM INDUSTRY ....	139
ANNEXE D: AN IMPROVED GENETIC ALGORITHM APPROACH FOR ON-LINE OPTIMIZATION PROBLEMS .....	164

# INTRODUCTION

Pour compenser des conditions changeantes sur la demande, les entreprises ont besoin de réviser constamment les priorités de fabrication. Dans l'optique de respecter les objectifs financiers, un certain nombre d'entreprises sont passées d'une mentalité de « réaction aux changements sur les prévisions » à un raisonnement à base de « stratégies adaptatives ». Une stratégie adaptative peut se définir ici comme une stratégie d'exécution qui permet de réviser les priorités de fabrication pour compenser des conditions changeantes dans l'optique de respecter les objectifs financiers et opérationnels. Dans ce contexte, les entreprises ont besoin de détecter les exceptions (bris de machine, déviation dans les temps opératoires, ordre urgent, etc.) de façon proactive, de résoudre ces difficultés en collaborant avec leurs fournisseurs et leurs sous-traitants, et d'en tirer les leçons pour améliorer les procédés en temps réel.

Les systèmes adaptatifs se trouvent à la croisée des chemins de deux domaines scientifiques majeurs : d'une part les systèmes d'information et d'autre part la recherche opérationnelle. Avec la démocratisation des progiciels de gestion intégrés (*Enterprise Resource Planning*, ERP) et l'essor des systèmes d'exécution de la production (*Manufacturing Execution System*, MES), les systèmes d'information tendent vers des applications plus complexes nécessitant des comportements plus intelligents. Les méthodes de la recherche opérationnelle tendent vers la résolution de problèmes plus réalistes pouvant demander des réponses en temps réel. La notion de temps réel n'est pas à prendre au sens absolu, mais le taux de réaction du système de commande doit correspondre au taux de changement de l'environnement contrôlé. L'aide à la prise de décision en temps réel n'est pas une question de système rapide, c'est plutôt une question de système suffisamment rapide pour agir sur l'environnement contrôlé de la façon désirée par l'entreprise.

D'un point de vue général, la difficulté de construire un système adaptatif vient des contraintes imposées par le système lui-même, sa tâche et son environnement. Le système est limité au niveau du traitement des données par sa vitesse et sa mémoire. Il est aussi limité au niveau de sa capacité de détection et d'action par ses capteurs et actuateurs, leur plage de fonctionnement, leur



précision, leur temps de réaction, etc. Néanmoins, ce système adaptatif doit accomplir une large variété de tâches allant des réactions d'urgence critiques pour l'environnement, à la résolution de problèmes plus complexes pouvant impliquer la recherche de solutions dans un espace de résolution de taille exponentielle. De plus, l'environnement impose lui-même une complexité et des contraintes dynamiques sévères.

Plus spécifiquement, un système adaptatif doit :

- travailler en continu sur de longues périodes de temps ;
- s'intégrer à l'environnement extérieur (atelier et autres départements de l'entreprise) ;
- traiter les incertitudes et les données manquantes ;
- concentrer les ressources sur les événements les plus critiques ;
- traiter aussi bien les événements synchrones qu'asynchrones en garantissant des temps de réponse acceptables ; et
- avoir des procédures de récupération en cas de défaillance du système de commande.

La réalisation d'un tel système ne peut pas être le seul fruit des avancées en système d'information ou en recherche opérationnelle, mais nécessite que les méthodes et techniques de ces deux disciplines soient combinées et intégrées. En effet, l'intégration insuffisante entre les ERP et les MES ne permet pas de garantir que les organisations seront capables de répondre rapidement aux événements qui surgissent dans l'atelier. D'autre part, l'une des différences clés entre un modèle hors-ligne et un système d'aide à la décision en temps réel se trouve au niveau des données. Le système décisionnel doit avoir accès aux informations critiques, particulièrement dans un environnement hautement compétitif et très changeant. Il doit aussi combiner différents outils de résolution.

Notre objectif principal est donc de proposer un modèle intégré de planification et d'ordonnancement réactif. Il devra nous permettre de répondre aux questions suivantes :

- Comment modéliser et résoudre ce problème de façon analytique dans le cas d'un processus adaptatif issu du milieu industriel ?
- Comment implanter l'approche de résolution proposée au sein d'un environnement applicatif industriel ?

Plus concrètement, le modèle visera l'implantation de composantes d'aide à la prise de décision en temps réel ainsi que des mécanismes de boucle rétroactive intégrant l'optimisation et les techniques de simulation au sein d'applications de gestion intégrée (ERP) et de gestion manufacturière (MES) permettant ainsi de connecter l'atelier de production au reste de l'entreprise. La plateforme qui sera mise en place permettra de répondre en temps réel aux divers aléas survenant dans l'atelier et pourra être étendue au-delà de la problématique de l'ordonnancement.

Pour y répondre, plusieurs objectifs spécifiques doivent être atteints :

- **Proposer** une architecture systémique générique permettant de détecter les événements perturbateurs et d'y réagir ;
- **Développer** des composantes d'optimisation en temps réel (algorithme) pour la planification et l'ordonnancement réactif issu du cas industriel ;
- **Proposer** des mécanismes d'aide à la prise de décision permettant au décideur de guider les composantes d'optimisation ; et
- **Développer, implanter et tester** un prototype d'outil d'aide à la décision en temps réel dans l'environnement issu du cas industriel.

Afin d'atteindre ces objectifs, la méthodologie suivante fut adoptée. Afin de proposer une architecture cohérente avec les réalités du marché et du monde industriel, il était important de faire la revue de la littérature sur les systèmes MES, ainsi que sur les progiciels d'exécution. Une emphase particulière a été mise sur l'évaluation de l'intégration entre ces deux systèmes et sur leur rapport avec des outils d'optimisation. Cette étape fut menée en partenariat avec STI Corp pour la partie MES et avec SAP pour la partie ERP. La connaissance pratique de ces outils fut

aussi un aspect fondamental de cette recherche. La formation sur l'ERP de SAP concernait aussi bien l'utilisation quotidienne du système que son paramétrage et les technologies d'intégration qu'il propose et qu'il supporte. STI Corp m'a aussi formé sur le MES Proficy® de General Electric. À l'issue de ces formations, nous avons développé une plateforme d'aide à la décision permettant de détecter les événements perturbateurs et d'y réagir. Cette plateforme est le résultat de l'intégration des systèmes ERP et MES avec un module d'aide à la décision. Des cas d'analyse ont été développés avec la participation de SAP et de STI Corp. Ces cas nous ont permis, d'une part, de faire évoluer la plateforme proposée et de nous assurer de sa viabilité et de sa tenue dans un contexte proche d'un environnement industriel. La suite du développement s'est concentrée sur le moteur de réordonnancement et sur son intégration au sein de la plateforme proposée. Ce moteur combine des heuristiques pour la génération de solution avec un simulateur en ligne pour leur évaluation.

En plus de cela, nous avons participé à un projet d'un an financé par SAP ayant pour but de développer un concept et un prototype d'un outil d'ordonnancement *lean*. Cet outil d'ordonnancement devait être conçu pour répondre aux besoins des petites et moyennes entreprises et devait avoir pour principales caractéristiques d'être rapide, interactif et facile d'utilisation. En plus du prototype, cela nous a permis de rencontrer une vingtaine de planificateurs dans diverses industries, de réaliser une cartographie de leur processus d'ordonnancement et de mesurer l'écart existant entre l'ordonnancement dans le monde industriel et dans le monde académique. Nous avons aussi effectué deux stages en milieu de travail chez STI Corp. Ces stages nous ont permis d'être formé sur un système MES, mais aussi de participer à la phase de développement d'une solution MES.

Cette thèse est structurée comme suit :

- Le chapitre 1 présente une revue de la littérature permettant de positionner la recherche effectuée par rapport aux travaux précédents dans les domaines des systèmes d'information et de l'optimisation en temps réel.

- Le chapitre 2 est une synthèse des travaux menés dans le cadre de cette recherche et constitue ainsi un fil conducteur cohérent des articles scientifiques présentés en annexe par rapport aux objectifs de recherche tant au niveau de la méthodologie que des résultats.
- Le chapitre 3 présente une discussion générale en regard des aspects méthodologiques et des résultats en lien avec la revue de la littérature.

La conclusion est suivie des articles en annexe sur lesquels se fonde cette thèse.

## **CHAPITRE 1: REVUE DE LA LITTÉRATURE**

L'objectif de cette revue de la littérature est de présenter le concept de production adaptative. Afin de le positionner par rapport à l'agilité, nous commencerons par définir ces deux concepts et mettre en lumière les aspects qui les rapprochent et ceux qui les différencient. Nous présenterons ensuite les défis et les difficultés industrielles que suscite la mise en œuvre d'une stratégie adaptative et nous terminerons par une revue spécifique aux quatre étapes de cette stratégie introduite lors de sa définition. Cette revue est orientée vers les deux principaux domaines de cette recherche : les systèmes d'information et l'optimisation en temps réel.

### **1.1 Production adaptative**

Par le passé, le monde de la production était gouverné par les économies d'échelle et tout bon gestionnaire savait que la production de masse et la pleine utilisation des capacités de production de l'entreprise étaient la meilleure façon de faire de l'argent. Ce type de gestion avait pour résultats une grande inflexibilité de l'outil de production et des difficultés de reconfiguration, le tout associé à des niveaux de matières premières, de produits intermédiaires et de produits finis élevés. Depuis le début des années 1980, l'industrie s'est lancée dans la recherche d'une plus grande flexibilité, l'élimination des inventaires excessifs, la réduction des délais de production, l'amélioration de la qualité aussi bien du produit que du service à la clientèle. Chez les analystes industriels, cela s'est traduit par la popularisation des termes « entreprise de niveau international » et « production lean » (Sheridan, 1993).

Depuis les années 1990, la compétition au niveau mondial, les changements technologiques ainsi que les demandes des clients ont créé pour les entreprises un environnement turbulent, complexe et incertain. Le problème sur la façon dont une entreprise peut répondre à des événements imprévus et imprévisibles dans un environnement dynamique changeant continuellement est devenu un sujet d'intérêt aussi bien pour les industriels que pour les membres de la communauté scientifique. De ces changements dans l'environnement est apparu le modèle de production agile (Goldman & Nagel, 1993; Goldman et al., 1995; Ismail et al., 2006; Vokurka & Fliedner, 1998; Yusuf & Adeleye, 2002; Yusuf et al., 1999; Zhang & Sharifi, 2000). Ce modèle fait le lien entre

les innovations au niveau de la production, l'information et les technologies de l'information avec une organisation radicalement repensée et de nouvelles stratégies de commercialisation (Gunasekaran, 1998). Considéré par certains auteurs comme une condition nécessaire de la compétitivité dans un futur proche (Goldman & Nagel, 1993; Hormozi, 2001; Sharifi & Zhang, 2001), c'est un modèle flexible capable de s'adapter rapidement aux changements de l'environnement d'affaire et pouvant répondre aux besoins croissants des clients. Il a de plus été considéré comme l'exigence ultime pour atteindre les performances d'une entreprise de niveau international (Hormozi, 2001), ou tout du moins l'étape indispensable vers le statut d'entreprise de niveau international (Yusuf & Adeleye, 2002).

Quand on demande de nommer le concept défini par la capacité de s'ajuster et de répondre au changement, les notions d'agilité, de flexibilité et d'adaptabilité sont les réponses les plus fréquentes et les plus populaires. Pour certains auteurs, la différence entre ces concepts est claire, mais d'autres les utilisent comme des synonymes (Sherehiy et al., 2007).

Dans cette section, nous présentons notre définition de ce qu'est une production adaptative. Étant donnée l'inévitable comparaison avec la production agile, il est important de commencer par clarifier les notions d'agilité et de flexibilité. Cela l'est d'autant plus que le manque de consensus autour du concept d'agilité, le flou entourant sa définition et sa compréhension impliquent bien souvent une confusion quand « agilité », « flexibilité » et « adaptabilité » sont utilisées conjointement.

## **1.1.1 Définitions**

### **1.1.1.1 Agilité**

Pour définir et comprendre ce qu'est l'agilité, deux approches sont à distinguer dans la littérature. La première approche consiste en un concept très large et imprécis qui englobe toutes les définitions et descriptions des pratiques et technologies variées ayant été implantées ces 20 dernières années (Goldman et al., 1995; Yusuf et al., 1999). Ainsi, cette conception de l'agilité est compatible avec des pratiques telles que la production lean, la productique (*Computer-*

*integrated Manufacturing*, CIM), la gestion intégrale de la qualité (*Total Quality Management*, TQM), la planification des ressources globales de l'entreprise (*Manufacturing Resource Planning*, MRP II), le juste à temps et les équipes autonomes. Dans la seconde approche, l'agilité est une adaptation rapide et proactive des éléments de l'entreprise à des changements inattendus et imprévus, et représente un modèle d'affaire nouveau et radicalement différent (Kidd, 1996; Sanchez & Nagi, 2001; Tsourveloudis & Valavanis, 2002). Étant donné que les pratiques compatibles avec la première approche présentent toutes des limitations bien connues quand elles sont appliquées dans un environnement aux changements incertains et imprévisibles, ces pratiques ne peuvent être incluses dans la seconde approche.

En plus de ces deux approches antagonistes sur le même concept d'agilité, d'autres problèmes apparaissent quand on cherche à distinguer l'agilité et la flexibilité. Les deux concepts sont définis de manière identique quand ils s'appliquent à l'entreprise au sens large, mais des différences apparaissent quand ils se réfèrent à la stratégie de production (Sherehiy et al., 2007). Dans ce cas particulier, le concept d'agilité peut être considéré comme une approche avec des pratiques, des techniques et des idées qui lui sont propres.

Ensuite, quand ces trois concepts sont appliqués à des domaines particuliers, les définitions changent et les différences entre eux évoluent aussi. Par exemple, l'agilité dans le domaine de la chaîne d'approvisionnement est définie comme étant la capacité à répondre rapidement et de manière adéquate à des changements à court terme de la demande, de l'approvisionnement ou de l'environnement (Charles et al., 2009). Cette définition caractérise un peu plus précisément la perturbation dans l'espace et dans le temps. La notion de rapidité reste présente. L'adaptabilité est utilisée dans un contexte à moyen terme et se différencie ici de l'agilité sur l'aspect temporel de la perturbation (Swafford et al., 2006). La flexibilité, quant à elle, est présentée comme un fondement de toute organisation agile (Christopher & Towill, 2000; Swafford et al., 2006).

Afin d'éviter ces écueils, des définitions communément acceptées s'imposent. Les définitions trouvées dans la littérature sont toujours dépendantes soit du contexte soit de l'échelle de l'entité étudiée, et apportent alors des divergences concernant les caractéristiques de l'entité agile, le

cadre conceptuel ou les outils de l'agilité. Comme ce fut le cas aux premiers jours de la logistique industrielle, nos travaux se baseront sur la définition militaire de l'agilité (adaptée au monde industriel). Cette définition est la plus précise que nous ayons trouvée sans pour autant être dépendante d'une échelle ou d'un contexte. L'autre avantage à utiliser une définition du domaine militaire est sa stabilité dans le temps : elle fige le sens de ce concept et le protège ainsi de tout phénomène de buzz. Dans ce contexte, l'agilité est la combinaison synergique des six attributs suivants, les six dimensions clefs de l'agilité (Alberts & Hayes, 2003) :

1. la robustesse : faculté de maintenir un niveau d'efficacité pour un large spectre de tâches, situations et conditions ;
2. la résilience : faculté à récupérer ou à s'adapter à la malchance, à un dommage ou à une perturbation déstabilisante de l'environnement ;
3. la réactivité : faculté à réagir à un changement de l'environnement en un temps approprié ;
4. la flexibilité : faculté à employer différents chemins pour arriver à ses fins et capacité à passer de l'un à l'autre aisément et de manière transparente ;
5. l'innovation : faculté à faire de nouvelles choses et faculté à faire d'anciennes choses d'une manière nouvelle ; et
6. l'adaptation : faculté à changer les processus de travail et faculté à changer l'organisation.

En s'appuyant sur cette définition, il est intéressant de noter que :

- cette définition est suffisamment générique pour pouvoir s'appliquer facilement à tout autre domaine que le domaine militaire ;
- l'efficacité est une hypothèse implicite et sera mesurée sur une dimension dissociée de l'agilité ;



- la rapidité n'est pas ici une fin en soi, mais un moyen d'arriver à sa fin : se mouvoir rapidement, mais pas de manière intelligente ne constitue pas ce qu'on peut appeler de l'agilité ;
- si ces six attributs sont analytiquement distincts, dans les faits ils sont souvent très interdépendants ;
- l'agilité ne doit pas être considérée comme étant un attribut purement relatif à une entreprise : l'agilité doit imprégner les personnes, l'organisation, les processus et la technologie ; et
- la relation entre l'agilité et la flexibilité est clairement établie.

### **1.1.1.2 Production adaptative**

Les lacunes des structures ou des organisations existantes et plus traditionnelles, mais aussi des employés, des matériels ou des technologies peuvent être identifiées en observant les capacités requises pour pouvoir évoluer dans un environnement instable et imprévisible. Quatre capacités essentielles sont ainsi requises :

1. la compréhension : faculté de comprendre la situation qui se dévoile ;
2. la collaboration : faculté de travailler dans un environnement collaboratif incluant les fournisseurs et les partenaires d'affaire, aussi bien sur le plan humain que technologique ;
3. la possession des moyens appropriés pour répondre à la situation ; et
4. l'orchestration : faculté d'orchestrer ces moyens au moment opportun.

La compréhension d'une situation commence par la mise en contexte de l'information disponible sur cette situation et par la recherche de schémas cognitifs connus et appropriés à l'enchaînement d'événements. Mais cela signifie aussi être capable de rendre l'information disponible à toute personne qui en aurait besoin, dans une forme utilisable, de manière sécuritaire et au moment opportun. La compréhension va au-delà de ce qui est en train d'arriver ou de ce qui pourrait arriver, elle va jusqu'à déterminer ce qui pourrait être fait pour remédier ou pour profiter de la

situation. Cela implique de générer des options, de prédire l'évolution de l'environnement (interne ou externe) et d'appréhender les effets d'une succession d'action.

La compréhension de la situation va aussi permettre de déterminer les moyens qu'il faudra posséder pour répondre. Cela aura des implications aussi bien sur les informations nécessaires que sur les acteurs qu'il faudra intégrer au plan et faire collaborer pour évaluer la situation et les actions à mettre en œuvre.

Pour qu'une collaboration soit effective, les participants doivent faire preuve d'un niveau suffisant d'interopérabilité dans le but d'échanger de l'information, de collaborer à la planification et au contrôle et pour bénéficier des effets de la synchronisation.

La faculté d'agir conjointement au moment voulu n'est pas la même chose que la vitesse de réponse. Au contraire, c'est une réponse au moment opportun. Néanmoins, la capacité d'agir collectivement et rapidement est importante puisque cela augmente la probabilité de pouvoir agir à l'instant approprié. Dépendamment de la complexité des décisions prises, il peut y avoir une certaine tension entre la faculté d'agir de manière synchronisée et la faculté d'agir rapidement.

**Ces quatre capacités essentielles forment le cœur de notre définition de ce qu'est une production adaptative.** Trois de ces capacités impliquent la manière de gérer les processus manufacturiers, la capacité restante est au sujet des outils. La production adaptative commande et contrôle des processus et des structures hautement adaptables. Cela signifie qu'ils doivent être modulaires, capables d'intégrer rapidement et efficacement de nouveaux acteurs, employer des boucles de rétroaction de manière naturelle et élémentaire, être capable de répondre très rapidement quand cela s'avère décisif, mais également de prendre le temps de développer une meilleure compréhension et de meilleures approches quand cela est possible, et aussi être en mesure de prévoir et reconnaître les changements de l'environnement interne et externe.

La production adaptative se base sur le paradigme d'adaptabilité où les cycles de décision sont incrémentiels et itératifs. Le temps d'un cycle élémentaire est la somme des temps requis pour détecter le changement et comprendre la situation, de décider des actions à prendre (ou de déléguer à d'autres cette prise de décision), et finalement de mettre en œuvre les actions décidées

(Figure 1.1). La vision des différents phénomènes est ici holistique et se concentre sur la dynamique et les effets des interactions. L'objectif n'est pas ici la poursuite absolue de l'optimalité, mais plutôt de garder la situation sous contrôle à l'intérieur de frontières permettant d'atteindre nos objectifs. Cela doit mener à l'amélioration des indicateurs de performance, ce qui sera validé par une continuelle confrontation du modèle avec la réalité.

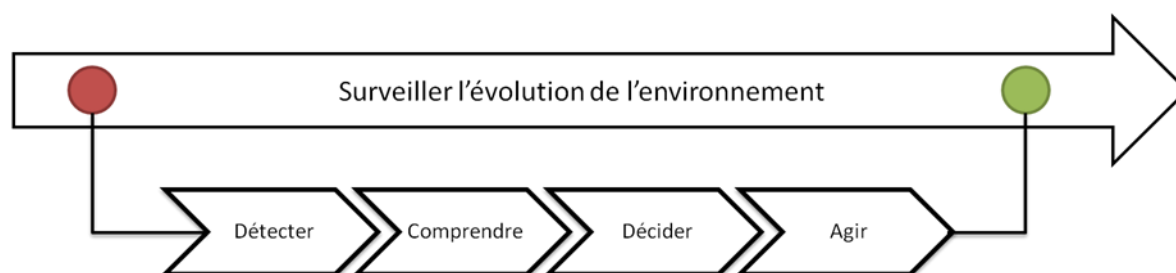


Figure 1.1 – Décomposition d'une stratégie de production adaptative

La production adaptative n'est pas la production agile pour deux raisons principale : d'une part ne couvre pas l'ensemble des six attributs définissant l'agilité (Figure 1.2) car elle est essentiellement concentrée sur l'horizon à court terme. Les capacités de robustesse, de réactivité et de flexibilité ont un impact immédiat sur les performances à court terme de l'entreprise, alors que la résilience, l'innovation et l'adaptation sont plus des capacités à moyen et long terme. **Il est donc possible de définir une production adaptative comme étant la combinaison de moyens robustes, réactifs et flexibles nécessaires pour réagir efficacement dans un environnement de production imprévisible et instable.** Une entreprise qui mettrait en œuvre le concept de production adaptative tel que décrit plus haut, le système de production de cette entreprise posséderait alors trois des attributs clés de l'agilité : la réactivité, la flexibilité et la robustesse. Mais comme les six attributs de l'agilité sont interdépendants, en accédant à la production adaptative, l'agilité n'en devient que plus facile à atteindre.

Comme pour l'agilité, la production adaptative doit imprégner tous les éléments de l'entreprise (employés, organisation, processus et technologie). Mais au-delà de cela, sans les bons moyens, sans les bons systèmes de production et d'information, la production adaptative devient

impossible à atteindre. Effectivement, au sujet des capacités de détection, de compréhension et d'orchestration, un système d'information pour la production adaptative doit être capable de contrôler et commander en temps réel les processus de production ou tout du moins il doit avoir une structure et des algorithmes d'ordonnancement capables de faire face aux incertitudes de l'environnement de production.

Une entreprise traditionnelle souhaitant se démarquer de la concurrence se doit d'offrir à sa clientèle une valeur ajoutée supplémentaire, que ce soit des délais de livraison plus courts, une plus grande personnalisation ou des prix moins importants. Ce sont les processus de production qui permettront d'atteindre ces objectifs. Dans un tel contexte, la production adaptative peut être décrite comme « connecter les machines aux marchés ». Le concept de production adaptative est capable de travailler avec le système physique de production existant, mais il est bien évident que les gains seront plus importants si l'adaptabilité de l'organisation est aussi transposée à l'atelier de production par l'utilisation de systèmes physiquement fortement adaptatifs tels que les systèmes de production reconfigurables (Koren, 2003).

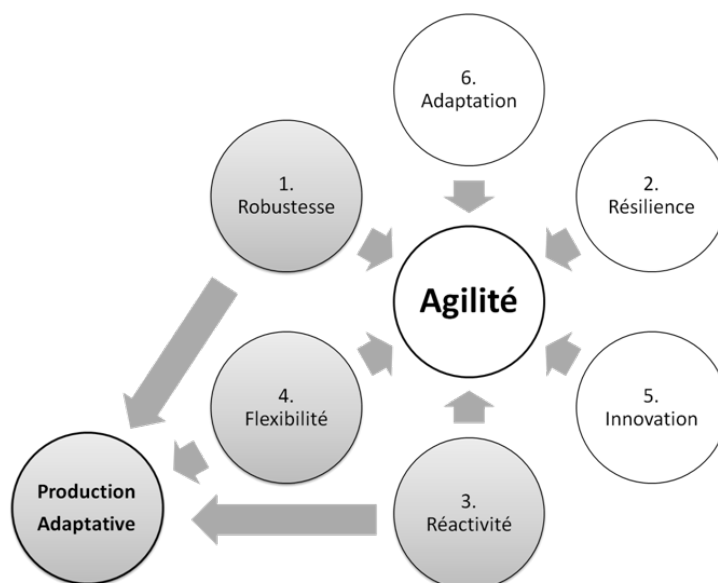


Figure 1.2 – Production adaptative vs. Agilité

### **1.1.2 Défis industriels et difficultés**

De nos jours, les industriels font face à des tendances connues sous les noms de réseautage, dynamisation et individualisation. Le terme de réseautage est à comprendre dans le sens de la coopération interentreprise. Grâce à son réseau, l'entreprise peut acheter sur le marché les composants dont elle a besoin, lui permettant alors de se concentrer sur son cœur de métier. Dans le cadre d'une stratégie de gestion de la chaîne d'approvisionnement, cela lui permet de s'intégrer efficacement dans la chaîne de production d'un produit fini. La dynamisation a pour origine les grandes fluctuations du marché qui, amplifiées par la grande quantité d'information disponible et leur rapide dissémination, encouragent les consommateurs à modifier rapidement ses habitudes de consommation. Les demandes de ceux-ci évoluent et l'entreprise est plus à l'écoute de leurs besoins, demandant plus de personnalisation des produits. La conséquence logique est une augmentation du nombre de variantes que l'entreprise doit offrir à ses clients.

Le réseautage, la dynamisation et l'individualisation favorisent une augmentation des risques et de la complexité au niveau des ateliers de production qui doivent être en mesure de s'adapter. Ces turbulences se caractérisent par de nouvelles exigences dans le traitement interne des ordres, mais aussi dans la dynamique avec les marchés externes. Cela se traduit par un réseautage externe plus important, par la collaboration avec de multiples partenaires, certains connus, d'autres nouveaux, et par des adaptations techniques et structurelles plus rapides. Ces nouvelles instabilités font qu'il est plus difficile pour la production de s'approcher de l'optimum économique, que la gestion de l'information devient inefficace et que les processus d'affaire deviennent inadaptés. Les conséquences pour le client sont alors un manque de fiabilité dans les livraisons, les délais d'obtention et la qualité du produit. Dans bien des cas, l'entreprise subira des délais de livraison trop longs qui, à leur tour, entraîneront des niveaux de stocks trop élevés. La liste des effets de ces turbulences est longue. Ces effets ont un impact à tous les niveaux de l'entreprise. Ils peuvent être atténués en créant plus de transparence entre et dans les différents départements, en améliorant les capacités de réaction et en sécurisant les dépenses. Pour y parvenir, l'entreprise va

être confrontée à deux problèmes majeurs : l'hétérogénéité du paysage industriel et la réactivité du processus décisionnel.

#### **1.1.2.1 Hétérogénéité du paysage industriel**

Les progiciels de gestion intégrée, aussi connus sous l'acronyme ERP (*Enterprise Resource Planning*), ont été conçus pour intégrer tous les processus d'affaires internes d'une entreprise. Néanmoins, les gestionnaires se plaignent souvent de l'inflexibilité des systèmes ERP causant des difficultés quand vient le temps de transposer dans la configuration du système les améliorations continues apportées à l'atelier (Akkermans et al., 2003). Le même point est rapporté par Upon et McAfee (2000). Ils relèvent aussi l'antagonisme des approches d'amélioration continue et de « big bang » inhérentes aux systèmes ERP actuels. Alors que les demandes des clients continuent de changer toujours plus rapidement que les processus d'affaires et la structure de la chaîne d'approvisionnement doivent s'adapter pour pouvoir suivre le mouvement, les systèmes ERP ne devraient pas freiner le processus d'innovation, mais plutôt l'accompagner.

Mais pendant qu'ils se concentraient sur l'implantation de systèmes ERP, les spécialistes des systèmes d'information ne donnaient que peu d'attention à l'atelier (Holst, 2001). Dans un tel contexte, il n'est pas surprenant que les départements de production aient de longue date favorisé le développement d'applications personnalisées pour remplir certaines fonctionnalités telles que le support aux opérations de production. Les systèmes de collecte de données de production basés soit sur des feuilles de calcul soit sur une base de données sont communément utilisés dans l'atelier pour surveiller et contrôler des processus d'exécution en temps réel. La maintenance des logiciels et la consolidation des données deviennent forcément complexes dans de tels environnements où le nombre et la structure de ces petites applications varient dans le temps.

Heureusement, les difficultés engendrées par l'intégration de ces systèmes ont poussé les fournisseurs de logiciels à rassembler ces composants multiples dans des solutions uniques et intégrées. Ces systèmes d'exécution de la production, communément appelés *Manufacturing Execution Systems* (MES), fournissent une interface utilisateur unique associée à un système de

gestion de données. Les fonctionnalités des MES sont nombreuses et peuvent supporter la gestion des opérations de production de la relâche des ordres en production à la livraison du produit fini (MESA, 1997b). En utilisant des données précises et actuelles, le MES peut guider les activités de production, réagir aux événements et suivre l'évolution de la situation.

Parmi les fonctionnalités des MES, certaines sont directement liées aux processus de production (ordonnancement, contrôle de la qualité), tandis que d'autres sont plutôt des fonctionnalités transversales (gestion des ressources, traçabilité). Cette difficulté à classer les fonctionnalités a deux conséquences. Tout d'abord, le concept de MES est souvent difficile à clairement délimiter pour les entreprises concernées. Ensuite, ces entreprises ont tendance à y mettre tout ce qu'elles n'ont pas réussi à assigner à d'autres systèmes tels que les ERP. La contribution majeure de la norme S95 (Instrument Society of America, 2000), publiée par le comité *Instrumentation, Systems & Automation* (ISA), est l'identification des interactions critiques entre les différentes parties d'un MES. Cette norme formalise aussi les échanges d'information entre le système de production et les autres secteurs de l'entreprise. Écrite pour être en mesure de s'appliquer quelque soit la stratégie de production, elle n'impose pas de modèle organisationnel ou une architecture particulière du système de production. Elle suggère un modèle physique extrapolé de la norme S88 (Instrument Society of America, 1995) et une définition des fonctions et des flux d'information basée sur le modèle de référence de l'Université Purdue (*Purdue University Reference Model*, PRM). Le modèle fonctionnel de la norme S95 met évidemment la fonction de contrôle de la production dans une position centrale. Le MES devient alors un intermédiaire entre l'atelier de production et les différents départements de l'entreprise. Pour une revue de littérature plus complète sur les MES, le lecteur peut se reporter à l'Annexe 1.

Cependant, les MES ne couvrent pas tous les processus permettant de replanifier la chaîne logistique pour répondre dynamiquement à un changement imprévu. Les stratégies adaptatives nécessitent aussi de définir la couche de gestion des processus entre l'ERP et le contrôle de la production et de traduire les besoins d'une chaîne logistique axée sur la demande en termes d'investissement en capacités, en systèmes et en intégration des processus d'affaire (Martin, 2004). Pour ce faire, les organisations manufacturières adaptatives ont besoin d'un MES qui

connecte les processus de production au reste de l'entreprise et de la chaîne logistique et qui fournisse une aide à la décision au personnel de l'atelier, leur permettant de livrer tout en respectant leurs objectifs de performance.

L'adoption de format et d'interface propriétaires pour les fournisseurs de logiciels reste la principale raison limitant l'intégration de ces différents systèmes. À cause d'eux, l'implantation, l'intégration et la maintenance des MES sont coûteuses. Les grandes entreprises manquent de temps alors que les petites manquent de ressources pour achever ces projets (Koch, 2001). Bien que le développement de solution MES soit une industrie prospère (Wildeman et al., 2008), les coûts élevés de développement ajoutés aux formats propriétaires freinent l'adoption de ces outils chez de nombreuses entreprises. Par sa nature et sa définition, les MES sont prêts à être intégrés à d'autres systèmes. Le problème vient du fait que chaque système d'information (ERP, SCM, etc.) a ses propres interfaces propriétaires et plus particulièrement les outils de production qui sont innombrables et bien souvent des modèles uniques. Les travaux académiques s'intéressant à ces problèmes d'intégration sont unanimes pour souligner l'importance d'une intégration du MES avec les autres systèmes d'information. Dès 1996, Westerlund (1996) insistait sur l'importance des capacités d'intégration dans le choix des systèmes ERP et MES, car c'est une des clés pour permettre aux entrepreneurs d'atteindre leurs objectifs financiers. En 2001, Rondeau et Litteral (2001) réitéraient cet appel. Liu et al. (2002) ont aussi décrit les difficultés d'intégrer un système de planification avancé (*Advanced Planning System*, APS), un ERP et un MES dans le but de pouvoir vérifier les contraintes de capacité des plans de production.

Une architecture orientée service (SOA, Service-Oriented Architecture) pourrait être une solution à ces problèmes d'intégration. SOA est une méthode de conception permettant de lier des applications ou des données à la demande pour arriver au résultat que le consommateur de service désire, ce dernier pouvant être l'utilisateur final ou bien un autre service. Ce type d'architecture promeut la réutilisation au niveau macroscopique (services) plutôt qu'au niveau microscopique (objets). Elle peut aussi grandement simplifier l'interconnexion aux systèmes d'information existants et leur usage. Le but de l'architecture SOA est d'obtenir un couplage « lâche » des interactions entre agents logiciels. Le service est une action exécutée par un « fournisseur » à



l'attention d'un « client ». L'interaction entre le fournisseur et le client se fait par le biais d'un médiateur responsable de la mise en relation des composants. Chaque application propose des services et met à disposition une description de ce service. Il suffit ainsi de mettre à disposition du médiateur tous les services disponibles et de les appeler quand cela est nécessaire. Cependant, cette solution nécessite la réécriture complète de toutes les applications actuelles (Erl, 2005). Mais aujourd'hui, de nombreuses recherches et des contributions majeures ont été présentées concernant les plateformes manufacturières orientées services (Chazalet & Lalanda, 2007; Jiang et al., 2007), mais le problème clé ici est le manque d'outils et de plateformes permettant de tester et de valider les développements de la recherche sur des problèmes réalistes aussi bien en terme de taille du système manufacturier que de l'ampleur des techniques de validation.

Howells (2000) met aussi en lumière la réalité des entreprises :

- indépendamment du produit fabriqué, un système de gestion financière peut être mis en place dans toute industrie ;
- indépendamment du produit fabriqué, les impératifs logistiques sont plus ou moins les mêmes ;
- c'est dans l'atelier que se trouvent les différences majeures.

En d'autres mots, les entreprises fabriquent des produits différents avec des équipements différents. C'est là la source de complexité de l'intégration de l'atelier. Actuellement, toutes les données sont transférées depuis des sources diverses par le biais de formats et de protocoles de communication variés.

### **1.1.2.2 Processus décisionnel**

L'ordonnancement des activités de production implique, pour chaque ordre de production et pour toutes les tâches correspondantes, de déterminer sur quels équipements ils seront effectués ainsi que les détails des dates de début et de fin de chacune des tâches tout en tenant compte d'un certain nombre d'objectifs et de contraintes. La littérature scientifique est riche en méthodes conçues pour optimiser certains critères de performance (profitabilité, temps de traitement total

d'une séquence, coûts, etc.) tout en faisant l'hypothèse que tous les paramètres sont déjà connus et ne changeront pas pour tout l'horizon de planification. En réalité, les ateliers sont rarement stables plus de trente minutes (McKay et al., 1988). Les contraintes évoluent, les cibles bougent. Les informations contenues dans les systèmes d'information ne sont bien souvent pas complètement à jour à cause des difficultés à traiter des données en temps réel. Par conséquent, la mise à jour de l'ordonnancement est effectuée manuellement par un planificateur humain. En connaissant la complexité des environnements de production, l'ordonnancement résultant est très souvent inefficace.

Les systèmes ERP montrent toujours certaines lacunes dans la gestion de la production. En premier lieu, ils héritent de la méthode de planification des besoins matières (*Material Requirements Planning*, MRP) de deux insuffisances majeures : l'hypothèse de capacité infinie pour les ressources et sa nature déterministe (Hopp & Spearman, 2000). De plus, les systèmes ERP nécessitent l'utilisation d'un système externe pour pouvoir collecter des données en temps réel (Mandal & Gunasekaran, 2002). Finalement, les systèmes ERP sont conçus pour collecter les traces d'événements, mais ne sont pas pensés pour aider le processus de prise de décisions (Moon & Phatak, 2005). Par conséquent, l'incapacité des systèmes ERP à traiter les incertitudes et les événements imprévus limite leur utilisation pour supporter le processus décisionnel dans un environnement de production dynamique.

Pour sa part, un MES doit être en mesure d'accéder aux données de l'atelier en temps réel pour pouvoir contrôler et commander les activités de production (MESA, 1997a, 1997b). Pour être réactif, il doit non seulement acquérir ces données, mais aussi les analyser en temps réel. Alors que le contrôle des systèmes manufacturiers se fait à l'échelle de la seconde, voire de la milliseconde, le MES peut prendre plusieurs minutes pour leur répondre. En comparaison avec la vieille image de l'atelier où les données étaient trop rarement mises à jour, ou bien encore quand les ERP étaient implantés en boucle ouverte et prenaient plusieurs jours pour répondre, on peut comprendre le désir d'opérer en « temps réel », mais il faut garder à l'esprit que si le système est plus réactif il n'est pas encore en temps réel au sens strict.

Généralement, l'extensibilité des MES se trouve surtout limitée dans sa capacité à coller à sa résolution de temps réel quand le système de production à contrôler voit sa taille augmenter. Il est clair que la quantité d'information collectée sur un système physique croît proportionnellement au degré d'automatisation de l'atelier. Autrement dit, le MES doit alors traiter plus d'information. Les systèmes de production voient leur taille augmenter à cause du besoin en processus plus complexe pour répondre aux nouvelles exigences des produits. En d'autres mots, le MES doit être physiquement et logiquement connecté à un plus grand nombre d'équipements et doit traiter proportionnellement plus de données. C'est un cercle vicieux : plus un MES veut être proche de la notion de temps réel, plus il doit être connecté ; plus il doit être connecté, plus il doit traiter de données ; finalement, plus il doit traiter de données, moins il est proche de la notion de temps réel.

En plus de cela, pour être en mesure d'améliorer la productivité du système de production d'une entreprise comprenant plusieurs sites, le MES doit pouvoir obtenir les données de tous les processus manufacturiers et sur tous les produits de toutes les usines géographiquement dispersées dans un temps respectable. Ces données devraient être analysées en temps réel pour prendre des décisions optimales globalement telles que la validation des paramètres d'un équipement, l'ordonnancement de la production, surveiller les statuts des processus, évaluer les tendances de la qualité des produits, et ainsi réagir plus rapidement aux demandes dynamiques des clients dans le cas d'un produit composé d'éléments provenant de diverses unités de production par exemple.

Jusqu'à présent, les fournisseurs de tels systèmes ont répondu à cette problématique en proposant différents modèles d'intégration entre les systèmes ERP et les MES. Même si ces modèles résolvent certains problèmes d'interopérabilité, ils n'apportent en revanche aucune capacité décisionnelle supplémentaire à aucun des deux systèmes.

## 1.2 Stratégie de production adaptative

Comme le montre la Figure 1.1, une stratégie adaptative est une succession de boucles Détecter, Comprendre, Décider, Agir. Le processus décisionnel est déclenché par un événement imprévu. Cet événement peut se manifester à l'intérieur ou à l'extérieur des limites physiques de l'entreprise. La première étape d'une telle stratégie est donc d'arriver à détecter automatiquement les événements qui intéressent l'entreprise. Dépendamment de l'événement et de son lieu d'apparition, il sera possible de le détecter par l'intermédiaire de différents systèmes. Par exemple :

- un système de gestion du cycle de vie des produits sera utile pour gérer l'évolution de la conception suite à la décision du client de modifier une partie du produit ;
- un système de gestion de la chaîne logistique sera capable d'envoyer une alerte si un des fournisseurs a des difficultés pour respecter ses dates de livraison ;
- un système ERP sera utilisé lors de la création, la mise à jour ou la suppression des ordres ; et
- un MES pourra détecter tout événement se produisant dans un atelier suffisamment automatisé (panne machine, retard de production, problème de qualité, etc.).

La deuxième étape correspond à la compréhension de la situation, l'identification de schémas connus. C'est une étape d'acquisition ou de sélection de données pertinentes et d'analyse. L'état le plus récent de l'atelier est nécessaire si l'on souhaite initialiser proprement la troisième étape correspondant au processus d'aide à la décision. Finalement, quand la décision a été prise, elle doit être propagée à chaque niveau de l'entreprise.

### 1.2.1 Détecter

Une stratégie adaptative se déclenche lorsqu'un événement imprévu survient. Cet événement peut intervenir soit à l'intérieur même de l'entreprise soit à l'extérieur (chez un fournisseur, un client, sur le marché, etc.). Les événements détectés ne sont pas n'importe quels événements. En effet,

ce sont ceux qui auront été mis en lumière lors de la conception des processus adaptatifs comme ayant un impact potentiel sur les processus de production ou d'affaire et auxquels l'entreprise souhaite pouvoir répondre soit pour parer à de potentielles difficultés, soit pour profiter d'une éventuelle opportunité. Saad et Gindy (1998) ont classifié ces perturbations en utilisant à la fois leur source et leur type (interne ou externe). Le Tableau 1.1 présente cette classification à laquelle nous avons ajouté le système ou la personne pouvant capter cette perturbation. L'information concernant les systèmes en mesure de détecter une perturbation est bien entendu conditionnée au paramétrage de ces outils par l'entreprise.

À cette liste, il faut en plus ajouter des perturbations dépendantes du cœur de métier de l'entreprise. Par exemple, une entreprise utilisant des moules en sable pour produire des pièces en alliage d'aluminium verra ses temps de séchage augmenter au printemps à cause de l'augmentation du taux d'humidité à cette période.

Les systèmes ERP utilisent un modèle transactionnel ce qui facilite maintenant le déclenchement d'événement dans ces systèmes corporatifs. Malheureusement, le problème peut toujours être présent au niveau de l'atelier, lieu où résident la diversité et la complexité. Cela a deux impacts. Premièrement, il est primordial de connecter les processus de production au système d'information adaptatif si l'entreprise souhaite être réactive aux événements de l'atelier ou pouvoir y propager ses décisions. Finalement, le système adaptatif doit être capable de s'adapter lui-même à une nouvelle organisation de l'entreprise tout en restant efficace.

#### **1.2.1.1 Connectivité et réseau**

La demande d'automatisation des processus de production est en croissance et de nombreux industriels font des efforts pour adopter ou améliorer les méthodes de communication et d'intégration de leurs équipements. Pour éviter les solutions propriétaires, les auteurs proposent d'utiliser des connecteurs génériques.

Tableau 1.1 – Perturbations, leur source et les systèmes impliqués dans leur détection

Source	Perturbation	Type	Détectée par
Machine	Indisponibilité (panne, maintenance)	Interne	MES
	Disponibilité, mais avec des limitations de capacité	Interne	MES
Outils	Indisponibilité (non réservé, partagé, mais utilisé par une autre ressource)	Interne	ERP / MES
	Disponibilité sous contraintes	Interne	MES / Superviseur
Transport	Indisponibilité (panne, etc.)	Interne	SCM
	Disponibilité avec une capacité limitée	Interne	SCM
Opérateur	Indisponibilité (maladie, vacances, etc.)	Interne	Superviseur
	Disponibilité sous contraintes (compétence manquante, etc.)	Interne	Superviseur
Demande	Ordres non prévus	Externe	ERP
	Ordres prévus retardés	Externe	ERP
	Ordres prévus en avance	Externe	ERP
	Changement de priorité des ordres	Externe	ERP
	Quantité plus/moins importante que planifiée	Externe	ERP
Fournisseur	Livraison à la mauvaise date	Externe	ERP / SCM
	Non-livraison des pièces requises	Externe	ERP / SCM

Yang, Cheng et al. (Cheng, Yang et al., 2000; Yang et al., 1999) ont développé un gestionnaire générique d'équipements capable aussi bien d'agir seul ou intégré à un MES. Pour atteindre l'objectif de généralité et de réutilisabilité, les règles gouvernant ce gestionnaire sont séparées en 3 portions indépendantes : les règles du système, les règles d'affaires et les règles de l'équipement. Dans leur première version, ces règles étaient implantées séparément dans

l'interface du MES et dans les pilotes des équipements. Les règles des équipements étant dépendantes de l'équipement, la connexion de machines différentes obligeait à réécrire le pilote. Les auteurs ont alors proposé d'utiliser la technologie d'objet mobile pour développer un pilote d'équipement accessible par internet. Le gestionnaire d'équipement télécharge le pilote par un navigateur et il peut dès lors communiquer avec l'équipement par les messages Java ou bien par la technologie des objets Internet avec les protocoles CORBA ou DCOM (Cheng, Lin et al., 2000).

Aussi, Ye et Qiu (Qiu, 2003; Ye & Qiu, 2003) ont défini un connecteur générique pour les équipements comme étant une plateforme capable d'être personnalisée à tout moment. Le support des différents protocoles de communication se fait par l'intermédiaire d'un composant de connectivité et d'une description des caractéristiques de l'équipement. L'utilisation d'un schéma basé sur la description des équipements permet d'assimiler des structures de données et des sémantiques différentes dans un contexte où les sources de données sont hétérogènes et permet ainsi de créer des mécanismes d'extraction d'information pertinente pour l'utilisateur.

Les systèmes contrôlés au travers d'un réseau se doivent d'intégrer les toutes nouvelles technologies telles que les réseaux sans fil, les systèmes embarqués, les terminaux mobiles et les tags électroniques (RFID), et de pouvoir ainsi répondre aux nouvelles exigences en matière de mobilité, de modularité, de contrôle et de diagnostic décentralisés ou distribués, d'autonomie et de redondance permettant une maintenance plus rapide et plus facile. Un des défis majeurs du domaine de la communication industrielle de ces architectures multi-niveaux est l'unification du réseau de l'entreprise grâce à la norme Ethernet. Le but ici est de garantir pour ce type de réseau les fonctionnalités des bus industriels existant dans l'atelier. Par exemple, l'application des techniques de tolérance ou de détection des erreurs aux systèmes de contrôle devrait améliorer la surveillance et le contrôle de ces systèmes complexes ainsi que leur fiabilité, leur sécurité de fonctionnement et leur capacité à adapter dynamiquement la performance du réseau, à reconfigurer ses composants et à atteindre la qualité de service désirée (Georges et al., 2006).

### 1.2.1.2 Architectures et modèles

Par sa connexion directe avec l'atelier, le MES joue un rôle important dans l'étape de détection des événements. La littérature scientifique a principalement proposé des modèles et des architectures de MES plus polyvalents et performants. Dans cette section, nous faisons la revue des développements devant faciliter leur intégration avec les systèmes existants. Nous discutons aussi de l'intégration entre MES et des efforts faits sur les modèles de données.

La complexité croissante des produits et des processus a remis en question la pertinence d'un outil totalement centralisé et a amené la proposition d'architectures à base d'agents. La technologie des agents a démontré sa simplicité d'implémentation et sa modularité (Shen & Norrie, 1999; Van Dyke, 1998). Les agents peuvent être représentés comme de simples ressources (une machine, une cellule, un outil), des opérateurs impliqués dans un processus, ou bien encore un produit, un client ou un fournisseur. Chacun des agents travaille à accomplir sa tâche dédiée tout en collaborant avec les autres dans le but de mener à bien les fonctions de production qui leur sont confiées (approvisionnement, planification locale, assignation des tâches, ordonnancement, contrôle de l'exécution, etc.). Une des applications les plus fructueuses de cette approche est le système manufacturier holonique (Valckenaers et al., 1994) où un holon est l'association d'un agent logiciel et d'un équipement ou d'un ensemble d'équipements. La coopération des holons ne se fait pas ici de manière centralisée ou décentralisée, mais elle reprend une hiérarchie basée sur une décomposition fonctionnelle. Il est possible de classer les architectures à base d'agents en deux groupes (Dilts et al., 1991; Leitão & Restivo, 1999) : les structures hiérarchiques où il y a plusieurs niveaux avec des relations du type maître/esclave et les structures hétérarchiques où les agents communiquent sur le modèle du partage, sans relation maître/esclave. Les agents sont localement autonomes et coopèrent au travers de procédures de négociation pour atteindre leur but. Les avantages de cette architecture résident dans sa complexité réduite, sa robustesse intrinsèque, son extensibilité, sa modularité et sa tolérance aux erreurs. Les décisions sont prises localement et uniquement quand cela est nécessaire.



Par exemple, le projet PABADIS (*Plant Automation BAseD on DIstributed Systems*) (Diep et al., 2003; PABADIS Group, 2001) a pour but de développer un MES basé sur une approche distribuée et décentralisée, implémenté grâce à des agents autonomes. Cette solution s'adresse d'abord aux unités flexibles produisant de petits lots, et devrait permettre aux entreprises de devenir hautement réactives sur un marché perturbé. Les holons sont aussi à la base d'une architecture MES adaptée à la tendance de la personnalisation de masse (Simao et al., 2006). SEMATECH (SEMATECH Inc., 1998) est une autre architecture ouverte pour l'intégration des MES pour l'industrie des semi-conducteurs. La version originale de cette architecture était surtout focalisée sur le contrôle de l'atelier. Des fonctionnalités telles que la conception de produit et la planification ont été ajoutées (Harhalakis et al., 1994, 1995). Cette architecture est accompagnée d'outils d'analyse, de conception et de documentation utilisant le langage UML et les réseaux de Pétri (Lin et al., 2005; Lin & Jeng, 2006).

L'exécution de la production est une tâche complexe à cause de la nature non linéaire du système de production, des processus manufacturiers et des incertitudes du système et de son environnement. Les plans et les cédules, provenant des niveaux supérieurs de l'organisation manufacturière, peuvent devenir infaisables en quelques minutes. De plus, la variété des types de systèmes de production existants et des problèmes de performance, ainsi que les différentes sortes d'équipement et de processus est très grande. Pour faire face à toute cette hétérogénéité, la conception des futurs MES doit appliquer les plus récentes et les plus fondamentales avancées faites dans le domaine des systèmes à organisation autonome, domaine intensivement étudié par la communauté des systèmes multi-agents (Di Marzo et al., 2004). Pour concevoir de tels systèmes, il est essentiel de s'appuyer sur la recherche fondamentale (Valckenaers et al., 2003; Waldrop, 1993) et de définir le cadre de modélisation adéquat permettant d'obtenir les fonctionnalités désirées. Un progrès fortement attendu dans le domaine est l'émergence de MES capable de prévoir l'évolution de l'état du système tout en conservant le niveau de découplage qui a rendu robustes et configurables les anciens MES basés sur les multi-agents (Valckenaers et al., 2004).

S'il est plutôt fréquent d'aborder le sujet de l'entreprise étendue au niveau des systèmes ERP, les articles traitant d'intégration prennent cependant rarement en considération les interactions entre MES, entre plusieurs ateliers d'une même entreprise ou encore d'une même chaîne d'approvisionnement. Or, le travail collaboratif devient de plus en plus courant, tout comme l'externalisation de certaines fonctions. Des termes tels que l'entreprise étendue, les organisations contractantes et l'entreprise virtuelle ont permis de décrire la coopération entre les entreprises. Mais ces alliances ont surtout été étudiées au niveau financier et rarement au niveau des ateliers. Huang (2002) propose un modèle d'intégration des MES entre eux basé sur les modes de coopération et les flux d'opération. Mais des limitations importantes apparaissent dues aux modèles de données que peuvent avoir chacun des MES ainsi intégrés.

### **1.2.2 Comprendre**

Les données et tout ce qui les entourent sont au cœur de l'efficacité des MES. Ils doivent analyser les données acquises et les mettre en contexte pour l'utilisateur. Comme le flux d'information circule avant le flux de matière, si l'information véhiculée est de mauvaise qualité et transmise lentement, il devient alors difficile d'assurer la réactivité et la qualité de la production. Il est évident pour tout le monde qu'avoir la bonne information au bon moment aide à prendre de bonnes décisions. Les technologies actuelles peuvent assurer la disponibilité de l'information, mais elles ne peuvent pas encore en garantir la précision et la signification (Jones et al., 2002). Si la précision d'une information dépend de la chaîne d'acquisition mise en œuvre, le sens qui lui est associé dépend uniquement de l'analyse que l'utilisateur en fait (Rockwell, 2006; Slater, 2000).

L'une des possibilités pour rassembler des données à des fins d'analyse est de les extraire directement d'une ou plusieurs bases de données opérationnelles. Les bases de données opérationnelles ou transactionnelles sont conçues pour exécuter rapidement et efficacement chaque transaction. Ce type d'interaction à base de transaction est connu comme le traitement transactionnel en ligne (*On-Line Transactional Processing*, OLTP). Quand ces données transactionnelles n'ont plus d'intérêt à être utilisées dans l'environnement de production, elles

peuvent être transférées dans un entrepôt de données pour être utilisées par des systèmes d'aide à la décision. Un modèle entité relation est souvent utilisé pour concevoir le modèle de données des systèmes OLTP. Cependant, les modèles en étoile ou en flocon (Wrembel & Koncilia, 2006) sont plus utilisés dans le cas des bases de données multidimensionnelles des entrepôts de données. Le traitement analytique en ligne (*On-Line Analytical Processing*, OLAP) est une méthodologie à base de requête supportant l'analyse de données dans un environnement multidimensionnel. Un moteur OLAP structure logiquement les données multidimensionnelles sous forme de cube. Par exemple, MS Excel fournit une interface permettant de visualiser de tels cubes. Ainsi, les cubes OLAP et les bases de données du MES peuvent être utilisés pour analyser les données de l'atelier. Chen et Wu (2005) ont créé une série de cube OLAP permettant l'analyse des performances des équipements, de leur encours et de la qualité de leur production.

Dans l'atelier, les données collectées par les équipements et les instruments de mesure ont le niveau de détail nécessaire au paramétrage des processus de production. Par exemple, ces données sont très souvent utilisées pour l'analyse statistique des procédés ou l'ajustement des paramètres du processus de production en cours (Xu & Qiu, 2004). Cependant, si un système d'information opérant à un niveau supérieur a besoin de ces données, elles seront trop souvent sans valeur pour l'utilisateur si elles ne sont filtrées et mises dans un contexte d'affaire significatif (Baliga, 1998; Qiu et al., 2002; Slater, 2000). Pour filtrer les données, il est possible d'utiliser les technologies de traitements de données standardisés telles que le modèle DOM (*Document Object Model*) (Qiu, 2002; Ye & Qiu, 2003). Les règles d'extraction dépendent de l'application, mais sont indépendantes des sources de données. L'utilisation du XML (XML Core Working Group, 2003) permet de rendre l'intégration dynamique et facilement extensible. Si en plus, il est combiné avec le langage XSLT (XSL Working Group, 2001), il devient alors possible de modifier dynamiquement la configuration (Xu & Qiu, 2004).

### **1.2.3 Décider – Modèles d'optimisation en temps réel**

L'une des fonctionnalités clés d'une stratégie de production adaptative tient dans l'approche d'optimisation en temps réel. Théoriquement, l'optimisation en temps réel implique des décisions

instantanées. Parce qu'une prise de décision complètement en temps réel est impossible en pratique, une approche réaliste doit être en mesure de proposer une solution faisable dans une plage de temps suffisamment courte pour être compatible avec le fonctionnement de l'atelier. Aujourd'hui, alors que la puissance de calcul des ordinateurs augmente, la commande de système en temps réel gagne en attention et génère un intérêt considérable aussi bien des communautés scientifiques qu'industrielles. Dans cette section, une revue des approches d'optimisation en temps réel les plus rencontrées est présentée. Sur la base des méthodes employées, ces approches peuvent être classifiées en cinq catégories (Figure 1.3) (Shukla & Chen, 1996).

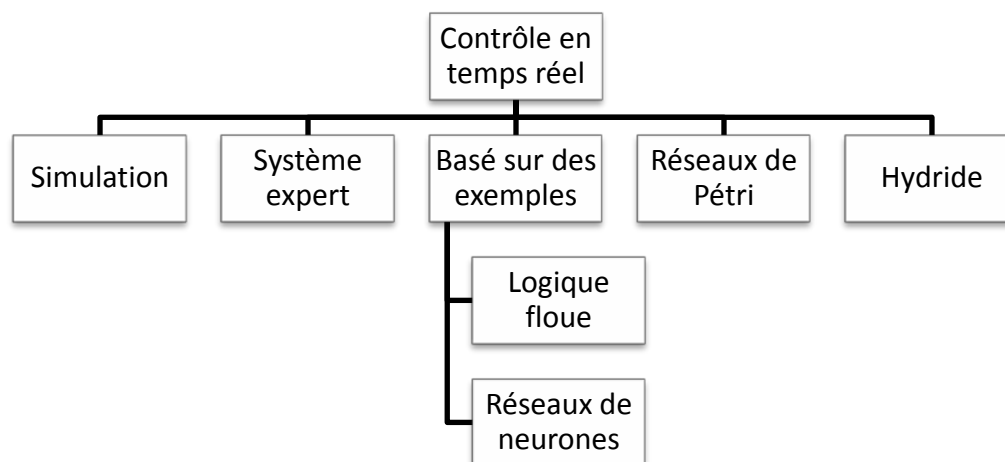


Figure 1.3 – Approches de contrôle en temps réel

### 1.2.3.1 Modèles basés sur la simulation

Dans cette section, une brève revue des approches utilisant la simulation pour la prise de décision en temps réel est présentée. Avant de présenter la revue de la littérature liée à ces approches, il est intéressant de rappeler l'importance de la conception du modèle de simulation pour une telle méthodologie. En fait, pour des approches classiques d'optimisation telles que la programmation mathématique, il est indispensable de connaître à l'avance la fonction de transfert. De plus, il est plus aisé de n'utiliser que des variables quantitatives dans ces processus d'optimisation. Ce n'est malheureusement pas le cas dans un contexte manufacturier stochastique. Ainsi, les modèles de

simulation sont une bonne alternative pour décrire le comportement dynamique et stochastique du système. Si les variables de décision sont les conditions sous lesquelles la simulation tourne, les mesures de performances sont une ou plusieurs réponses données par la simulation. Dans la littérature, les approches d'optimisation basées sur la simulation peuvent être classifiées en six catégories (Figure 1.4). Dans la suite de cette section, nous nous concentrerons sur la plus répandue des catégories dans la prise de décision en temps réel, les méthodes heuristiques. Pour plus de détail sur les autres méthodes, le lecteur peut se référer à (Carson & Maria, 1997).

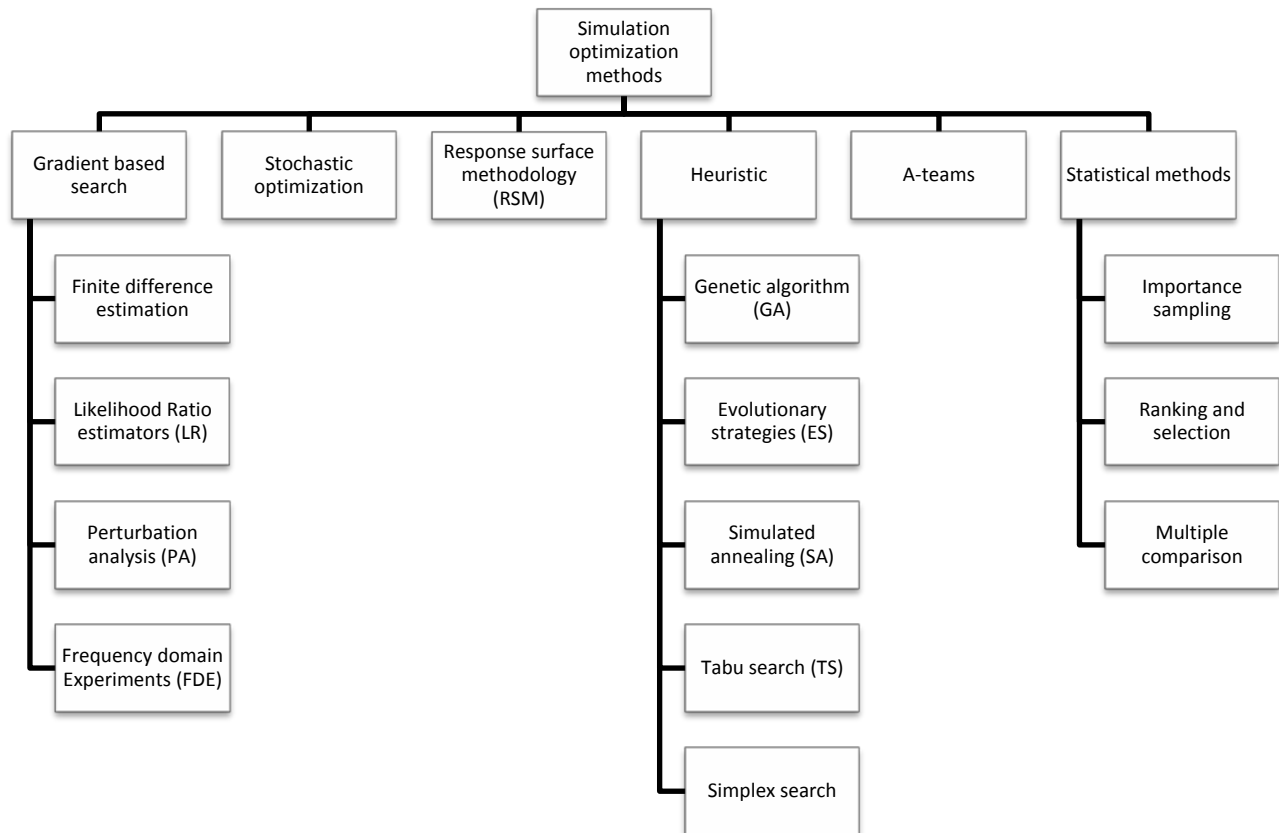


Figure 1.4 – Méthodes d'optimisation basées sur la simulation

Les méthodes heuristiques consistent en une exploration aléatoire des solutions admissibles de l'espace de décision. Le processus de recherche s'achève quand une condition d'arrêt est remplie. À chaque point du processus de recherche, la valeur de la fonction objectif du problème est estimée par le modèle de simulation. Ainsi, aucune information sur la forme analytique de la fonction objectif n'est requise. Cette catégorie couvre la méthode du simplexe (Azadivar & Lee, 1988), la recherche tabou, le recuit simulé (Lee & Iwata, 1991; Ogbu & Smith, 1990), les algorithmes génétiques et les stratégies évolutionnaires. Il est intéressant de noter que de précédentes recherches et revues de littératures (Azadivar & Tompkins, 1999; Chaudhry & Luo, 2005; Ruiz et al., 2007) ont démontré l'efficacité des algorithmes génétiques et des stratégies évolutionnaires. Les systèmes basés sur la simulation ont été employés dans plusieurs études pour résoudre des problèmes d'ordonnancement et de contrôle en temps réel. Elmaraghy et al. (1998) proposent ainsi une méthode combinant l'optimisation et les techniques de simulation capable de s'attaquer aux problèmes d'ordonnancement en ligne. Dangelmaier et al. (2006) développe un système prédictif-réactif assisté par la simulation pour l'ordonnancement et le réordonnancement de systèmes de production flexible types. Les résultats obtenus ont montré que les questions de stabilité et de contrôle en temps réel peuvent être appréhendées efficacement en combinant la simulation avec l'optimisation.

Il est intéressant de noter que l'état de l'art sur l'optimisation et la prise de décision en temps réel basée sur la simulation a été couvert par de nombreuses publications de qualité. Nous renvoyons le lecteur vers Boesel et al. (2001), Chan et Chan (2004), Yoon et Shen (2006) pour des revues récentes de la littérature. Au regard du futur de telles approches, les experts pensent que l'utilisation de la simulation dans le domaine manufacturier est en nette augmentation.

### **1.2.3.2 Modèles basés sur un système expert**

Cette classe de systèmes est basée sur la connaissance empirique du système et utilise des règles, des heuristiques ou la connaissance d'un expert pour réduire le processus de recherche (Gonzalez & Dankel, 1993). Dans la littérature, un grand nombre de systèmes experts pour les systèmes manufacturiers ont été développés. Aytug et al. (1994) et Shukla & Chen (1996) présentent une

revue de ces développements. Récemment, Choi et Kang (2005) ont fourni un outil flexible exploitant la connaissance du système pour prendre des décisions en ordonnancement. Gebus (2006) a proposé un système d'aide à la prise de décision, utilisant un système expert, pour l'optimisation de la production et l'amélioration de la qualité dans l'industrie électronique. Ils démontrent la faisabilité de telles approches au travers de plusieurs cas d'étude. Comme le montrent plusieurs recherches, le plus sérieux problème de ces systèmes experts ou à base de connaissances vient du manque de moyens pour mettre à jour automatiquement les connaissances elles-mêmes. Ce point limite l'usage de telles approches dans le contexte du réordonnancement.

### **1.2.3.3 Modèles basés sur la logique floue et les réseaux de neurones**

Cette classe de modèles utilise les situations passées pour aider la prise de décision actuelle. Basés sur des scénarios types et sur l'intelligence artificielle, ces systèmes cherchent à prendre une décision en répétant ou en améliorant une décision du passé. La logique floue, comme les systèmes à base de connaissances, essaie d'automatiser le raisonnement humain. Les deux systèmes diffèrent dans la façon d'implémenter les règles (Shukla & Chen, 1996). Les réseaux de neurones sont des collections de modèles mathématiques qui émulent certaines propriétés des systèmes nerveux biologiques et leur faculté d'apprentissage. Comme le mentionnent Tang et al. (2005), le principal avantage des réseaux de neurones réside dans leur résistance contre les distorsions dans les données d'entrée et dans leur capacité d'apprentissage. Ces caractéristiques donnent à ces systèmes de bonnes perspectives dans le domaine de l'ordonnancement dynamique. Nous renvoyons le lecteur vers Tang et al. (2005) pour connaître les récentes applications de cette méthode pour résoudre des problèmes d'ordonnancement dans un contexte dynamique. Dans le même contexte, Li et al. (2005) ont abordé le problème de l'ordonnancement d'atelier multi-gamme avec un réseau de neurones flou, combinant ainsi ces deux méthodes. Les résultats montrent que l'algorithme d'ordonnancement en temps réel est efficace comparé à d'autres méthodes.

#### **1.2.3.4 Modèles basés sur les réseaux de Petri**

Les réseaux de Pétri, comme outils graphiques et mathématiques, fournissent un environnement uniforme pour la modélisation, l'analyse formelle et la conception de systèmes à événements discrets (Zurawski & Zhou, 1994). Les réseaux de Pétri sont reconnus comme étant des outils de modélisation très utiles pour les systèmes présentant des phénomènes de simultanéité, de synchronisation et aléatoires. Depuis leur apparition, de nombreux types de réseaux de Pétri ont été développés pour répondre à différents problèmes. Ils ont été utilisés dans plusieurs études pour résoudre des problèmes d'ordonnancement et de contrôle en temps réel. Chong et Chen (1998) utilisent un réseau de Pétri coloré et un concept d'optimisation ordinal pour choisir rapidement la politique d'ordonnancement la plus efficace dans un contexte d'atelier flexible. Ils rapportent que même si l'approche ne peut pas garantir la meilleure performance globale, elle fournit des performances satisfaisantes pour le cas étudié. Reyes et al. (2002) utilisent les réseaux de Pétri et une intelligence artificielle basée sur des méthodes heuristiques pour étudier la conception et l'ordonnancement de systèmes manufacturiers flexibles et montrer l'efficacité de la méthode dans ce contexte. Toujours dans le même contexte, Mejia et Odrey (2005) utilisent les réseaux de Pétri et une heuristique améliorée pour aborder un large éventail de problèmes d'ordonnancement. Ils ont appliqué leur approche au problème classique d'atelier multi-gamme ainsi qu'à des problèmes complexes d'ordonnancement de systèmes manufacturiers flexibles. Leurs résultats sont prometteurs en termes de qualité de la solution trouvée et de temps de calcul.

#### **1.2.4 Agir – Propagation de la décision**

Quand une décision a été prise, elle doit être propagée vers les personnes ou les systèmes impliqués par le processus de prise de décision ou impactés par la décision. Toutes ces personnes ne travaillent pas nécessairement dans le même département et n'interagissent pas avec les mêmes systèmes (ERP, MES, etc.). Pour propager efficacement l'information, les systèmes d'information doivent être convenablement intégrés.



Le MES a été créé pour combler le vide entre l'atelier et les systèmes de planification (Koch, 2001; Owen & Parker, 1999; Slater, 2000). Dès lors, l'intégration de ce système dans le réseau des systèmes d'information de l'entreprise est cruciale pour rendre les données collectées en temps réel de l'atelier (stocks disponibles, utilisation des équipements, séquence de production en cours, etc.) accessibles en temps voulu, car une connexion efficace de l'atelier n'est pas une option, mais bien la pierre angulaire d'un système d'entreprise à l'échelle de l'entreprise et hautement intégré (MESA, 2000). Les bénéfices d'une telle intégration seront différents d'une entreprise à l'autre, mais les apports généraux se feront aux niveaux de la traçabilité, de la visibilité des processus et la possibilité de les rendre plus réactifs. Toutes les organisations manufacturières disposent d'une solution d'intégration, mais dans la plupart des cas elle est basée sur le papier ou sur le personnel (Adshead, 2007). Les modèles d'intégration sont de deux types : l'intégration logicielle (grâce aux interfaces de programmation ou aux agents) et l'intégration par les données.

Encore aujourd'hui, la plupart des systèmes d'information ne sont malheureusement pas conçus pour travailler au niveau de l'atelier. Howells (2000) observe d'ailleurs que les entreprises commencent par implanter les applications financières et poursuivent avec celles des ressources humaines, de la logistique, du marketing, de la chaîne d'approvisionnement, et seulement après tentent d'ajouter à l'ensemble un système lié à l'atelier. À cause du manque de considération de l'intégration au moment de la conception des premiers systèmes implantés, la connexion avec l'atelier se fait difficilement, au moyen d'interface de programmation (API) qui rend le développement coûteux, le système non flexible et difficilement maintenable (Koch, 2001; Linthicum, 1999; Slater, 2000).

Après le développement de solutions à grande échelle, non ouvertes et propriétaires, la définition d'interfaces telles que les BAPI de SAP R/3 est devenue une méthode d'intégration courante dans les systèmes hétérogènes. En raison de la nécessité à combiner les meilleurs modules de différents fournisseurs, le concept de composants prédéfinis réutilisables a été développé et peut être appliqué pour intégrer différents systèmes au sein d'une même entreprise ou bien entre organisations désirant développer leurs coordinations (Mustafa & Mejabi, 1999). L'utilisation de

ce genre d'interface permet de s'intégrer dans les processus internes du logiciel. Par exemple, les modèles de maintenance qui font l'objet de nombreuses recherches sont souvent intégrés aux ERP. Ip et al. (2000) ont conçu grâce à la méthodologie IDEF un système de maintenance s'intégrant dans la logique MRPII d'un ERP. Alors que Nikolopoulos et al. (2003) ont préféré l'approche objet, Rizzi et Zamboni (1999) ont développé un module intégré à un ERP permettant d'optimiser le stockage de matières périssables.

Il faut aussi noter que les API ne sont pas l'unique méthode d'intégration. Comme dans le projet PABADIS, l'architecture d'un MES peut aussi s'articuler autour d'agents mobiles. Mais les agents, grâce à leurs capacités à négocier de manière autonome pour réussir à atteindre le but qui leur est assigné, peuvent aussi être utilisés comme vecteur d'intégration. Ils ont déjà été implémentés pour gérer la capacité d'un atelier depuis un ERP (Mertens et al., 1994) ou pour incorporer formellement le contrôle de l'atelier dans un ERP et permettre le réordonnancement à la volée du cheminement des produits ou la reconfiguration des processus manufacturiers (Tang, Y. & Qiu, 2004). Le développement de cette méthode est facilité par les capacités du langage de programmation JAVA qui implémente nativement les agents mobiles (Wong, D. et al., 1999).

Comme nous le mentionnions plus tôt, toutes les données sont transférées depuis des sources diverses par le biais de formats et de protocoles de communication variés. Une intégration efficace des systèmes d'information passe donc d'abord par une bonne intégration des données. Par exemple, K. Wong (2006) propose un modèle de données permettant l'intégration d'un ERP et d'un système de gestion du cycle de vie des produits (*Product Lifecycle Management*, PLM). Avec la norme S95, l'intégration des différents niveaux de l'entreprise est plus facile à atteindre puisque le volume 5 (Instrument Society of America, 2007) s'intéresse aux transactions entre un ERP et un MES alors que le volume 6 (Instrument Society of America, 2009) s'intéresse à celles entre le MES et l'atelier. Cette intégration s'est longtemps faite par le biais de fichiers CSV (*Comma-Separated Values*) ou de chiffrier et n'était pas du tout flexible. Maintenant, la connexion entre un MES et un ERP est plus standard. Par exemple, des serveurs OPC (*OLE for Process Control* ou *Open Process Control*) peuvent rendre les données du système commandé accessibles et les formater au format B2MML (*Business To Manufacturing Markup Language*,

l'implémentation XML de la norme S95) pour les envoyer à l'ERP (Adshead, 2007). Le standard XML est l'approche idéale quand il faut harmoniser les données : l'intégration devient dynamique, extensible et rapidement reconfigurable (Salam et al., 1999).

### **1.3 Conclusion**

Les stratégies de production adaptative se trouvent à la croisée des systèmes d'information et de la recherche opérationnelle au sens large. Elles se distinguent toutefois par leurs aspects industriel, appliqué et applicable.

Les environnements instables et imprévisibles, à l'image des ateliers de production, sont le champ d'action privilégié d'une telle stratégie. Un système adaptatif devra posséder à la fois la capacité de comprendre la situation, de collaborer et d'orchestrer sa réponse. Il devra aussi posséder les moyens appropriés pour répondre. Les cycles de décision se décomposent en quatre étapes, à savoir, la détection de la perturbation, la compréhension de la situation, la prise de décision ou le support à la prise de décision et finalement la mise en œuvre des décisions prises.

Lors de la mise en place d'une stratégie adaptative, l'entreprise sera confrontée à deux défis majeurs. Elle devra d'abord faire face à l'hétérogénéité de ses systèmes d'information et à la diversité de son atelier de production. Elle sera ensuite confrontée à la réactivité de son processus décisionnel. En effet pour qu'une stratégie adaptative ait un sens, le taux de réaction du système adaptatif doit correspondre au taux de changement de l'environnement.

Dans le prochain chapitre nous présentons notre problématique, notre démarche ainsi qu'une synthèse de l'ensemble des articles présentés en annexe et sur lesquels s'appuie cette thèse.

## CHAPITRE 2:      SYNTHÈSE

Ce chapitre présente une synthèse de l'ensemble du travail de recherche et son articulation autour des quatre articles publiés dans des revues scientifiques. Il débute par une présentation de la problématique abordée et de la démarche utilisée. Il met en lumière les relations des différents articles entre eux et avec les objectifs spécifiques de la méthodologie. Par la suite, pour chaque article, une section reprend les aspects méthodologiques ainsi que les résultats obtenus.

### 2.1 Problématique

Comme nous avons pu le voir dans la revue de la littérature et en nous basant aussi sur l'offre existante, les MES actuels suivent à la lettre la définition du MESA (MESA, 1997b) : « Le MES fournit les informations nécessaires à l'optimisation des activités de production allant de la création de l'ordre de fabrication au produit fini. » Le MES alerte, présente et met en forme les données désirées par l'utilisateur... et il s'arrête là.

Un système d'aide à la prise de décision se base non seulement sur le traitement de l'information, mais aussi sur la recherche opérationnelle et les méthodes analytiques. La contribution de ces systèmes doit se faire sur chacune des six étapes du processus de prise de décision (partie inférieure de la Figure 2.1) : (1) stocker les données élémentaires, (2) récupérer les données, (3) sélectionner les données pertinentes, (4) concevoir le modèle de décision, (5) évaluer les décisions, (6) sélectionner et choisir une solution.

La Figure 2.1 présente les étapes d'une stratégie adaptative (partie supérieure) combinées au processus de décision. Les activités grisées sont déjà supportées par les MES. On voit ainsi que le MES, pierre angulaire d'une telle approche, ne couvre pas tous les processus permettant de replanifier la chaîne logistique pour répondre dynamiquement à un changement imprévu. Mais l'intégration du MES avec les autres systèmes d'information va rendre possible l'adoption de stratégie manufacturière adaptative. D'une part, cela renforce le besoin d'adopter les approches de conception orientée services ou multi-agent. D'autre part, les stratégies adaptatives requièrent de définir la couche de gestion des processus entre l'ERP et le contrôle de la production et de

traduire les besoins d'une chaîne logistique axée sur la demande en termes d'investissement en capacités, en systèmes et en intégration des processus d'affaire (Martin, 2004). Pour ce faire, les organisations manufacturières adaptatives ont besoin d'un MES qui connecte les processus de production au reste de l'entreprise et de la chaîne logistique et qui fournit une aide à la décision au personnel de l'atelier, leur permettant de livrer tout en respectant leurs objectifs de performance.

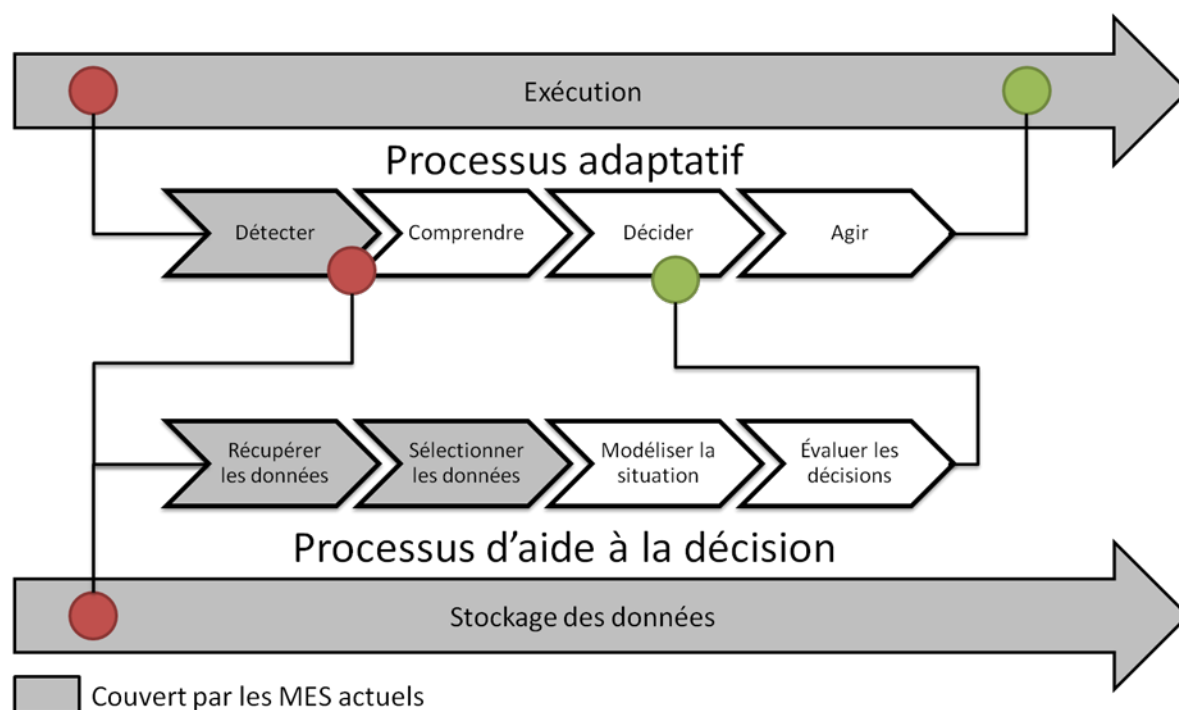


Figure 2.1 – Système manufacturier adaptatif intelligent

Sur la Figure 2.1, les activités en blanc sont non couvertes par les MES. Il est intéressant de se demander si ces activités sont couvertes par d'autres systèmes. Les ERP possèdent effectivement une fonction de planification et de contrôle de la production, basée sur la logique MRPII. Ils héritent de cette méthode deux insuffisances majeures : l'hypothèse de capacité infinie pour les ressources et sa nature déterministe (Hopp & Spearman, 2000). Dépendamment des problèmes d'ordonnancement de l'entreprise, la solution peut être suffisante, mais dans de nombreux cas

elle ne le sera pas. De plus, Akkermans et al. (2003) nous rapportent que les industriels perçoivent les systèmes ERP comme non flexibles face au monde en perpétuelle évolution que nous connaissons. De plus, les ERP favorisent une hiérarchie organisationnelle descendante et s'adaptent mal aux environnements de production en juste à temps. Ils deviennent même alors un frein aux politiques de progrès permanent. Les ERP ne semblent donc pas être la meilleure solution pour remplir l'activité d'ordonnancement détaillé.

Quand il s'agit d'optimisation et de planification, le système le plus apte semble être l'APS. Ces systèmes contiennent des fonctionnalités de planification de la production et d'ordonnancement (Stadtler & Kilger, 2000). Stadtler (2005) dresse une liste de problèmes et de défis pour les APS. La diversité des ateliers de production n'est pas prise en compte, les modèles d'optimisation sont déterministes, les incertitudes mal prises en compte et certaines sources d'incertitudes sont ignorées. Il montre aussi un intérêt pour les agents logiciels pour répondre au problème de décentralisation. Il aborde la question de l'intégration avec l'atelier et propose de connecter directement l'atelier et l'APS afin d'éviter une couverture fonctionnelle redondante avec les MES. De mon point de vue, s'il existe des redondances fonctionnelles entre ces deux systèmes, elles se trouvent plus au niveau des noms des fonctions que du contenu. Les fonctions des MES sont plus orientées vers la production et la mémorisation des données alors que l'APS est avant tout un outil de calcul. Malheureusement, si ces capacités d'optimisation peuvent s'avérer intéressantes pour des horizons de planification à long ou moyen terme, ou pour de l'ordonnancement prédictif, les défauts précédemment cités ne font pas des APS dans leur version actuelle de bon candidat pour couvrir les activités d'optimisation en temps réel.

Du point de vue des méthodes de réordonnancement en temps réel, la plupart des recherches ont compris l'intérêt d'utiliser des données représentatives de l'état réel de l'atelier, mais le lien avec les MES ne semble pas avoir encore été fait. Ainsi, Yoon et Shen (2006) présentent de nombreux outils décrivant différentes façons de collecter et d'utiliser les données de l'atelier, mais aucune ne fait mention des MES. La recherche en réordonnancement permet de combler en partie le vide entre la théorie de l'ordonnancement et les pratiques industrielles (Vieira et al., 2003). Les auteurs rapportent ainsi que les modèles mathématiques ne décrivent pas explicitement les

politiques qui vont effectivement commander le système. Ces politiques nuisent aux performances du système et le modèle du système est alors inadéquat. De la même façon, les indicateurs de performance utilisés pour juger de la performance d'un système doivent être mieux modélisés : si la performance financière d'une unité de production peut être un indicateur pour les décideurs, ce n'est certainement pas la seule. Les auteurs remarquent aussi un manque de recherche pour comparer les performances d'un même système manufacturier sous des politiques de réordonnancement différentes et pour comprendre les interactions entre les différentes fonctions de planification et l'impact de ces politiques sur la performance du système.

En plus de la difficulté de développer des modèles explicatifs réagissant en temps réel à des pannes imprévues et d'intégrer des traitements stochastiques, certains problèmes restent inexplorés :

- la collaboration entre le planificateur, le responsable de l'ordonnancement détaillé et le MES est virtuellement inexplorée ;
- le développement de la technologie MES dans le champ de la coordination et du contrôle entre plusieurs sites n'en est actuellement qu'aux prémices.

## **2.2 Démarche et objectifs de recherche**

Nous rappelons ici notre objectif principal qui est de proposer un modèle intégré de planification et d'ordonnancement réactif. Il devra nous permettre de répondre aux questions suivantes :

- **Q1** : Comment modéliser et résoudre ce problème de façon analytique dans le cas d'un processus issu du milieu industriel ?
- **Q2** : Comment implanter l'approche de résolution proposée au sein d'un environnement applicatif industriel ?

Plus concrètement, le modèle visera l'implantation de composantes d'aide à la prise de décision en temps réel ainsi que des mécanismes de boucle rétroactive intégrant l'optimisation et les techniques de simulation au sein d'applications de gestion intégrée (ERP) et de gestion

manufacturière (MES) permettant ainsi de connecter l'atelier de production au reste de l'entreprise. La plateforme qui sera mise en place permettra de répondre en temps réel aux divers aléas survenant dans l'atelier et pourra être étendue au-delà de la problématique de l'ordonnancement.

Pour y répondre, plusieurs objectifs spécifiques doivent être atteints :

- **OS1 : Proposer** une architecture systémique générique permettant de détecter les événements perturbateurs et d'y réagir ;
- **OS2 : Développer** des composantes d'optimisation en temps réel (algorithme) pour la planification et l'ordonnancement réactif issu du cas industriel ;
- **OS3 : Proposer** des mécanismes d'aide à la prise de décision permettant au décideur de guider les composantes d'optimisation ;
- **OS4 : Développer, implanter et tester** un prototype d'outil d'aide à la décision en temps réel dans l'environnement issu du cas industriel.

Comme nous le précisons en introduction, la problématique abordée se trouve à la croisée de deux domaines de recherche : l'optimisation et les systèmes d'information. La jonction de ces domaines est rendue nécessaire par l'aspect temps réel des stratégies adaptatives à mettre en œuvre et la volonté de prendre en compte l'état réel du système de production. Si certaines de nos contributions peuvent être plus orientées vers l'un de ces deux domaines, la majorité reflète l'orientation bicéphale de notre recherche. Les articles sur lesquels s'appuie cette thèse sont les suivants :

- **Article 1** : Saenz de Ugarte, B., Artiba, A., & Pellerin, R. (2009). Manufacturing execution system – a literature review. *Production Planning & Control*, 20(6), 525–539.
- **Article 2** : Saenz de Ugarte, B., Hajji, A., Pellerin, R., & Artiba, A. (2009). Engineering Change Order Processing in ERP Systems: An Integrated Reactive Model. *European Journal of Industrial Engineering*. (Accepté)



- **Article 3** : Saenz de Ugarte, B., Hajji, A., Pellerin, R., & Artiba, A. (2009). Development and integration of a reactive real-time decision support system in the aluminum industry. *Engineering Applications of Artificial Intelligence*, 22(6), 897-905.
- **Article 4** : Saenz de Ugarte, B., Artiba, A., & Pellerin, R. (soumis en octobre 2009). An Improved Genetic Algorithm Approach for On-line Optimization Problems. *Production Planning & Control*.

L'article 1 fait une revue de la littérature des MES. Ce système est la clé de la communication entre les systèmes d'information et les ateliers de production. Cet article est une exploration préliminaire à la proposition d'une architecture systémique (OS1). Les articles 2 et 3 présentent un modèle d'ordonnancement réactif (OS2) s'articulant autour d'une architecture d'exécution (OS1) en temps réel intégrant un algorithme génétique couplé à un modèle de simulation, avec un ERP et un MES appliqué aux industries de l'aéronautique et de l'aluminium. L'article 4 se place dans la lignée des deux articles précédents et propose deux mécanismes améliorant l'algorithme génétique utilisé pour de l'optimisation en temps réel et pouvant être étendus à d'autres heuristiques utilisant une fonction d'évaluation sous certaines conditions (OS2). Les sections suivantes présentent la synthèse de chacun de ces articles que le lecteur trouvera en version complète en annexe.

## 2.3 Article 1 – Revue de la littérature sur les MES

Depuis leur apparition au milieu des années 1990, les MES ont fortement évolués pour devenir des applications plus puissantes et mieux intégrées. Leur couverture fonctionnelle a aussi changé pour arriver à supporter la plupart des processus d'exécution manufacturier en un seul et même système. Cet article commence par une présentation formelle du concept de MES tel que définit par la MESA (*MES Association*) et par la norme S95. Il présente aussi l'état actuel des MES disponibles sur le marché, ainsi que les technologies qu'ils embarquent aujourd'hui pour répondre à la couverture fonctionnelle définie par la norme S95. Il s'intéresse particulièrement aux problèmes majeurs auxquels les MES sont confrontés du point de vue de leur architecture, de

la connectivité aux machines et aux réseaux, et de leur capacité de traitement des données. Les modèles d'intégration entre les MES et les ERP proposés par la communauté scientifique sont aussi présentés.

Mais ces outils ne sont pas non plus parfaits et présentent un certain nombre de limitations. Sur le plan architectural, si la communauté scientifique a proposé un certain nombre de nouveaux concepts, les outils manquent pour pouvoir les tester sur des problèmes et dans des conditions réalistes. L'article pointe aussi la difficile relation entre la taille du système physique contrôlé et la notion de temps réel de ce contrôle. En matière d'intégration, le MES est aussi placé dans un environnement hostile avec d'un côté les ERP, champion des technologies de communication propriétaires, et de l'autre l'atelier, champion de la diversité.

Malgré toutes ces avancées, nous constatons que les MES n'embarquent toujours pas d'outils d'aide à la décision qui permettraient la mise en œuvre de stratégies d'exécution adaptative réagissant à des événements imprévus et ainsi de mieux répondre aux changements d'un environnement dynamique. En effet, les contributions scientifiques sont potentiellement nombreuses, allant du développement de nouveaux algorithmes de réordonnancement jusqu'à la proposition de nouveaux indicateurs de performance sur l'adaptivité des processus. Les outils d'aide à la prise de décision en temps réel sont importants et se doivent d'intégrer des méthodes d'évaluation. À ce titre, la simulation reste un outil puissant pour évaluer différents scénarios et calculer des indicateurs pour les différentes stratégies considérées. La combinaison de la simulation en ligne et des algorithmes d'optimisation est maintenant relativement bien établie dans le monde scientifique, mais reste à accomplir dans le monde industriel.

L'intégration des MES avec les systèmes ERP et un module d'aide à la prise de décision semble donc prometteuse et c'est que les articles 2 et 3 vont permettre d'explorer.

## **2.4 Articles 2 et 3 – une plateforme d'exécution en temps réel réactive et intégrée**

Comme l'article 1 le montre, le MES propose un système unifié permettant de suivre l'exécution des processus de production, de la relâche des ordres de production jusqu'à la livraison des produits finis. Malgré tout, le MES n'est pas suffisant pour permettre de mettre en œuvre les stratégies adaptatives.

Pour combler ce manque, nous proposons dans cet article une plateforme d'exécution manufacturière générique où des modèles d'optimisation en temps réel s'intègrent au traditionnel tandem ERP/MES. L'objectif principal de cette architecture est de fournir un outil d'aide à la prise de décisions en temps réel flexible et intégré, réagissant à des événements imprévus surgissant soit dans l'environnement de production (panne, etc.) soit au niveau corporatif (nouvel ordre, etc.). Les différents systèmes peuvent être sur des machines différentes connectées à un même réseau. La connexion avec l'atelier utilise la norme OPC et l'interface entre l'ERP et le MES respecte la norme S95 et utilise des fichiers B2MML. Le noyau d'aide à la décision est un système hybride combinant un outil d'optimisation avec un modèle de simulation. Il se déclenche sous l'action d'un des systèmes surveillant l'apparition d'événements imprévus. Il s'initialise alors avec les données de l'ERP et du MES pour être le plus fidèle possible à l'état actuel du système. La simulation est ici utilisée pour l'évaluation des solutions. Finalement, le noyau renvoie des messages aux autres systèmes représentant les décisions prises. Cette architecture peut facilement être étendue pour intégrer d'autres systèmes et pouvoir ainsi réagir à de nouveaux types d'événements.

Pour démontrer la faisabilité de l'approche, l'architecture est d'abord mise dans un contexte de support de traitement des ordres de modification technique (ECO, *Engineering Change Order*) dans un environnement contrôlé par un ERP, puis dans le cadre d'une aluminerie cherchant à minimiser les déclassements de matières en fonction des propriétés chimiques de sa production.

### 2.4.1 Application démonstrative dans l'industrie aéronautique

Cette démonstration est basée sur un scénario réel de l'industrie aéronautique. Quand un ECO arrive, l'entreprise déclenche une action de remplacement immédiat. Si la pièce requise n'est pas en stock ou en cours de production, elle doit être produite le plus rapidement possible. Dans les ateliers considérés, les outillages sont partagés entre plusieurs ateliers suivant un calendrier établi longtemps à l'avance. Le planificateur doit donc replanifier sa séquence de production de sorte que l'ECO soit produit rapidement tout en minimisant le temps de production de cette séquence. L'ERP va donc être le déclencheur de la stratégie adaptative. Le noyau d'optimisation est composé d'un algorithme génétique et d'un modèle de simulation en ligne. L'ERP (SAP R/3) fournit les ordres à réordonnancer et l'ECO à l'algorithme génétique. L'algorithme génétique envoie les séquences à tester au simulateur qui s'initialise avec les statuts réels de l'atelier et les calendriers des outillages fournis par le MES (SAP MII). Le simulateur retourne les résultats de la séquence testée à l'algorithme qui poursuit son processus d'évolution. L'algorithme génétique est codé dans MatLab et le simulateur utilisé est développé en utilisant le langage Visual SLAM. Le noyau d'optimisation teste ici 3 scénarios :

- Scénario 1 : l'ECO est placé en début de séquence et les autres ordres gardent le même ordre que précédemment : c'est la solution pratiquée actuellement dans l'entreprise ;
- Scénario 2 : l'ECO est placé en début de séquence et les autres ordres sont reséquencés ;
- Scénario 3 : les autres ordres et l'ECO sont reséquencés ;

Au final, le planificateur sélectionne la solution qui lui convient, l'ajuste le cas échéant et la confirme. Le modèle a été validé au préalable sur un ensemble de problèmes de *jobshop* tirés dans la littérature. Le scénario 3 obtient les meilleurs résultats concernant la minimisation du temps global de production, mais il ne sera pas forcément choisi, car l'ECO peut parfois n'être produit que tardivement. Toutefois, en étudiant la réaction du modèle pour différents niveaux de la contrainte de disponibilités des outillages, on se rend compte que l'écart entre les scénarios 2 et 3 est peu significatif.

Comme l'étude porte sur une plateforme d'exécution en temps réel, l'étude des performances du noyau d'optimisation est importante. Cette performance est mesurée comme étant la somme du temps requis pour obtenir les données nécessaires au processus d'optimisation et renvoyer la réponse, du temps de calcul utilisé par l'algorithme génétique et du temps de simulation. Ces deux derniers temps dépendent fortement des paramètres de l'algorithme génétique (taille de la population, nombre de générations, etc.). Le comportement du système en fonction de ces paramètres est étudié en utilisant le calendrier d'outillage le plus contraignant. Sur les plages de paramètres étudiés, ces temps de calcul peuvent être multipliés par 5 pour un gain sur la fonction objectif minime. Ainsi si les paramètres de l'algorithme ont un impact faible sur la fonction objectif, c'est tout le contraire sur les performances du noyau d'optimisation.

La plateforme d'exécution manufacturière générique se montre flexible et le noyau d'optimisation est pleinement intégré aux systèmes d'information existants. Le cas illustratif montre la viabilité de la plateforme dans un mode de fonctionnement en temps réel. Il a aussi permis de mettre en lumière les différents aspects de l'intégration, l'utilité des différents éléments pour résoudre des problèmes de prise de décision en temps réel et l'impact des paramètres de l'algorithme d'optimisation sur les performances du système.

## **2.4.2 Application démonstrative dans l'industrie de l'aluminium**

Cette démonstration est basée sur un scénario réel de l'industrie de l'aluminium. Le processus de fabrication d'alliage d'aluminium est une combinaison de processus continus et discrets. L'aluminium est le résultat d'une électrolyse qui dissout le minerai (processus continu). Quand un bain d'électrolyse est plein, il est vidé dans un creuset (processus discret) qui sert alors à transporter l'aluminium jusqu'aux fours. Dans ces fours, l'alliage est préparé et mélangé avant d'être coulé en lingot. Dans notre cas, la ligne est constituée de 240 bains d'électrolyse vidangés toutes les 24 heures. Chaque bain a des propriétés chimiques qui lui sont propres et une heure de vidange spécifique. Les propriétés chimiques peuvent changer dans le temps et c'est le laboratoire qui est responsable des tests. Selon son niveau de contamination, l'aluminium sera utilisé dans des alliages de plus ou moins bonne qualité et donc de plus ou moins grande valeur

marchande. Le métal arrive aux fours en une séquence incontrôlée qui dépend des heures de vidange des bains et de la qualité de l'aluminium. Au niveau de la fonderie, le planificateur doit optimiser la cédule de production quotidienne en tenant compte des conditions de l'atelier en temps réel. Son but est donc de produire les alliages prévus tout en minimisant l'utilisation d'aluminium de bonne qualité pour produire des alliages de moindre qualité (déclassement).

Les ordres de production pour la semaine sont gérés par le système ERP. Le planificateur n'utilise pas d'outil dédié à la planification et met à jour la cédule 2 fois par semaine. Les ordres sont relâchés par le système ERP et envoyés en production. Un MES est utilisé pour la traçabilité des statuts de production et de la qualité de l'aluminium produit. Quand un ordre est partiellement ou totalement produit, le MES envoie une confirmation à l'ERP. Actuellement, la qualité des décisions prises dépend grandement de l'expérience du planificateur et du temps alloué à la création de la cédule.

La plateforme proposée est utilisée dans ce contexte. Elle est légèrement modifiée de sorte que toutes les communications entre les systèmes ERP, MES et d'aide à la décision transitent par un serveur d'intégration (SAP MII ici). De cette manière, chaque application n'a qu'un seul et unique intermédiaire. La plateforme gagne alors en flexibilité puisque l'ajout d'un système supplémentaire à la plateforme ne nécessite de connecter le nouvel entrant qu'à une seule application pour pouvoir communiquer avec l'ensemble des applications déjà en place. Le noyau d'aide à la décision est toujours composé d'un algorithme génétique et d'un simulateur. L'utilisation du serveur d'intégration a été poussée un cran plus loin puisque l'algorithme génétique et le simulateur ne communiquent pas directement entre eux, mais uniquement via celui-ci. Cette solution faisant une utilisation intensive des web service permet ainsi de changer l'algorithme générant les séquences sans avoir d'impact sur le simulateur. De la même façon, le simulateur peut être changé sans avoir à toucher à l'algorithme d'optimisation. L'algorithme génétique est toujours codé dans MatLab, mais le simulateur utilisé maintenant SDBuilder. Ce choix est principalement justifié par les capacités de communication native de SDBuilder.

La stratégie réactive d'ordonnancement se déclenche après chaque confirmation d'ordre, autrement dit quand un four se libère et qu'il doit être réaffecté. Par rapport à la cédule de référence fournie par le planificateur, l'utilisation de la plateforme a permis de diviser par 2 l'indicateur de déclassement de l'aluminium tout en réduisant de 10 % le temps de production de la cédule.

Comme pour l'article précédent, les performances de la plateforme ont été étudiées. Les temps de calcul de l'algorithme génétique et temps de simulation dépendent toujours fortement des paramètres de l'algorithme. Toutefois ici, le simulateur a été conçu autour de 3 modèles distincts : un contrôleur communicant avec l'algorithme génétique et recevant les séquences au fur et à mesure de leur génération, un modèle communicant avec l'atelier et se synchronisant à intervalle régulier pour refléter l'état de l'atelier et un modèle simulant le fonctionnement de l'atelier et servant à évaluer les séquences. La logique métier est uniquement dans ce dernier modèle, ce qui facilite grandement la maintenance du simulateur. Malgré tout, les paramètres de l'algorithme ont toujours un impact relativement faible sur la fonction objectif, mais important sur les performances du noyau d'optimisation.

Dans cet article un effort particulier a été fait pour accentuer davantage la flexibilité de la plateforme d'exécution en temps réel. L'utilisation d'un serveur d'intégration permet de limiter l'effort de développement lors de l'intégration d'un nouveau système à la plateforme existante. L'utilisation de web service a aussi permis de rendre interchangeable les blocs élémentaires du noyau d'optimisation.

Une des particularités du modèle de simulation utilisé était le grand nombre de ressources mises en jeu. Cela s'est traduit par une dégradation du temps d'évaluation des séquences proposées par l'algorithme génétique. Les capacités d'archivage des transactions du serveur d'intégration nous ont permis de pouvoir analyser à posteriori le comportement du noyau d'optimisation et de proposer des mécanismes d'amélioration de ce noyau.

## 2.5 Article 4 – Mémoire et mécanisme de prédation

Dans le domaine de l'optimisation en temps réel, toute perte de temps inutile peut être vue comme une pénalité à double tranchant. D'une part, le temps passé n'a servi à rien et ne pourra plus être récupéré et d'autre part, c'est une opportunité manquée d'améliorer la solution au problème. Dans cet article, nous traitons du cas où la fonction d'évaluation de l'algorithme génétique est consommatrice de temps. En nous fondant sur l'analyse des transactions émises par le noyau d'optimisation lors de nos précédents travaux, nous proposons deux mécanismes permettant soit de réduire la consommation de temps de la procédure d'évaluation sans dégrader les qualités de convergences, soit d'améliorer la rapidité de convergence à temps de calcul constant et tout en étant facilement intégrables aux heuristiques basées sur les algorithmes génétiques.

La première amélioration consiste en une mémoire. En effet, en analysant les séquences générées puis évaluées lors de nos expérimentations, nous nous sommes rendu compte qu'entre 10 et 30 % des solutions évaluées l'avaient déjà été précédemment. Autrement dit, entre 10 et 30 % du temps de calcul était gaspillé à cause de l'amnésie du noyau d'optimisation. L'ajout d'une mémoire peut se faire de deux façons. Dans la première implémentation, la mémoire se place juste avant le processus d'évaluation. Avant d'évaluer une solution, la mémoire est scrutée. Si la solution y est déjà, le résultat est directement renvoyé à l'algorithme génétique. Sinon, la solution est normalement évaluée, son résultat est mis en mémoire et retourné à l'algorithme. Cette implémentation permet ainsi de réduire le temps de calcul sans modifier le comportement de l'algorithme génétique et l'algorithme génétique n'a pas besoin d'être modifié. Dans la seconde implémentation, la mémoire est scrutée lors du processus de génération des solutions. Si les solutions générées sont déjà en mémoire, l'algorithme va réitérer le processus de génération. Ainsi, chaque solution évaluée est unique au travers des générations et la mémoire est donc intergénérationnelle. Le temps de calcul sera identique à celui d'un algorithme génétique sans mémoire, mais la convergence devrait être meilleure puisque le nombre de solutions différentes évaluées est plus grand dans le cas avec mémoire. Par contre, cette utilisation de la mémoire



nécessite de pouvoir modifier l'algorithme génétique. Dans les deux cas, l'utilisation d'une mémoire ne se justifie que si le temps moyen requis pour scruter la mémoire est inférieur au temps d'évaluation de la solution.

Le second mécanisme proposé s'insère directement dans le processus d'évaluation et n'est fonctionnel que sous certaines hypothèses. L'idée ici est de relaxer l'évaluation d'une solution avant son terme à partir du moment où il devient clair que cette solution est mauvaise. Les deux hypothèses de fonctionnement sont les suivantes :

- L'évaluation de la fonction objectif se fait pas à pas ;
- La fonction objectif est non décroissante d'un pas à l'autre (dans le cas d'une minimisation).

Le mécanisme proposé se compose d'une proie, la solution à évaluer, et d'un prédateur caché dans le processus d'évaluation et interrogé à chaque pas. Il est constitué de trois paramètres : une valeur par défaut renvoyée quand l'évaluation d'une solution est relaxée, un seuil de déclenchement et d'une probabilité de relaxation. Le fonctionnement du prédateur est le suivant. La valeur de fonction objectif pour la solution évaluée évolue pas après pas. Quand cette valeur devient supérieure ou égale au seuil de déclenchement, l'évaluation est relaxée avec une probabilité de relaxation. Si l'évaluation est relaxée, la valeur par défaut est renvoyée à l'algorithme, sinon l'évaluation se poursuit normalement et le prédateur n'agira plus sur cette solution. Si le seuil de déclenchement n'est pas atteint, c'est que la solution évaluée est bonne et il est alors possible de mettre à jour les paramètres du prédateur.

Afin de valider ces deux mécanismes et d'essayer de déterminer les conditions dans lesquelles leur apport est le plus conséquent, une série de tests et de comparaisons a été effectuée. Il a été ainsi trouvé que la mémoire est plus intéressante quand les chromosomes sont de petites tailles ou que le nombre d'évaluations est élevé. Cela est dû d'une part à l'opérateur de croisement qui fait souvent revenir les mêmes chromosomes quand l'algorithme converge et d'autre part, à la diversité combinatoire moins élevée quand les chromosomes sont petits. La mémoire intergénérationnelle semble aussi plus efficace quand l'algorithme ne génère pas une bonne

solution lors des premières itérations. Cette dernière situation sera plus fréquente si la taille de la population initiale n'est pas trop importante. Concernant le mécanisme de prédation, son efficacité semble plus importante sur les petits chromosomes. Cela est dû au nombre de fois où le seuil de déclenchement est testé. Cette limitation peut toutefois être aisément levée en adaptant le mécanisme à la taille du chromosome. Finalement, il était prévisible que de plus la probabilité de prédation est élevée, plus le mécanisme est efficace.

Ces résultats n'étant toutefois que d'ordre qualitatif, les deux mécanismes ont été testés sur le cas tiré de l'industrie de l'aluminium. Ce cas a été préféré à l'autre, car le temps nécessaire à l'évaluation d'une séquence était le plus important (de l'ordre de 5 secondes par solution évaluée). Dans le scénario de base servant de référence, le noyau d'optimisation est composé d'un algorithme génétique sans mémoire et d'un simulateur sans prédateur. Les paramètres de l'algorithme génétique sont réglés afin d'obtenir le meilleur compromis entre le temps de calcul et la qualité de la meilleure solution trouvée. Le simple ajout d'une mémoire intergénérationnelle va permettre de faire un gain de 10 % sur la valeur de la fonction objectif sans diminuer le temps de calcul.

Un premier prédateur est ajouté au simulateur. Sa valeur par défaut et sa probabilité de relaxation sont constantes. Le seuil de prédation utilise la meilleure valeur de la fonction objectif trouvée augmentée de 15 %. Le comportement du prédateur est étudié pour différentes valeurs de probabilité de prédation. On trouve alors que le temps de calcul diminue quand cette probabilité augmente. C'était un résultat attendu puisque plus la probabilité est élevée, plus le prédateur a de chance de relaxer l'évaluation. Toutefois, il n'y a aucune corrélation entre cette probabilité et le gain obtenu sur la fonction objectif, mais en moyenne le gain est d'environ 20 %. Malheureusement, ce prédateur donne de très mauvais résultats dans les cas où une très bonne solution arrive au début du fonctionnement de l'algorithme génétique. Dans ces quelques cas que nous avons observés, le seuil de prédation devient très bas trop tôt, ayant pour conséquence la relaxation prématurée de trop nombreuses solutions. Pour pallier ce défaut, le second prédateur implémenté sera moins actif au début de l'expérimentation. Le seuil de déclenchement utilise toujours la meilleure valeur de la fonction objectif, mais augmentée cette fois d'une marge

diminuant avec le nombre de fois où le prédateur relaxe une évaluation. Ce nouveau prédateur permet de réduire d'environ 4 % le temps de calcul par rapport au premier prédateur et d'environ 8 % la fonction objectif. Cela vient principalement du fait que le prédateur étant plus calme en début d'expérimentation, l'algorithme génétique a plus d'opportunité d'explorer l'espace de solutions.

Ainsi, les deux mécanismes proposés permettent, dans les cas où le processus d'évaluation consomme beaucoup de temps et sous certaines hypothèses, à la fois de réduire cette consommation de temps de l'ordre de 20 %, mais aussi d'améliorer aussi la fonction objectif d'environ 20 %. L'inconvénient majeur est l'introduction de nouveaux paramètres pour le prédateur, mais les expérimentations ont montré que ces paramètres n'ont pas besoin d'être réglés avec trop de finesse. Si ces deux mécanismes ont été développés à l'origine pour l'algorithme génétique, ils peuvent toutefois être utilisés avec n'importe quel autre heuristique utilisant aussi un processus d'évaluation.

## 2.6 Conclusion

Ce travail de recherche apporte des éléments de réponses à la problématique de planification et d'ordonnancement réactif en temps réel dans un milieu industriel. Plus particulièrement, le modèle proposé intègre des composantes d'aide à la décision en temps réel dans une architecture comprenant un ERP, un MES et tout autre système. La plateforme d'exécution en temps réel ainsi constituée permet de répondre en temps réel aux divers aléas survenant dans l'atelier. Elle sert donc de socle à la mise en place de processus adaptatif dans l'entreprise et pourra être étendue au-delà de la problématique de l'ordonnancement. La plateforme a été validée sur deux cas démonstratifs issus de cas réel de l'industrie de l'aéronautique et de l'aluminium. À partir de ces expérimentations, des mécanismes ont été proposés afin d'améliorer le comportement des algorithmes génétiques dans un contexte d'optimisation en temps réel avec un processus d'évaluation des solutions temporellement long. Ces mécanismes peuvent être utilisés par d'autres heuristiques dont le processus d'évaluation souffre de la même carence.

## **CHAPITRE 3: DISCUSSION GÉNÉRALE**

Ce chapitre présente une discussion générale de certains aspects méthodologiques et de nos résultats en lien avec la revue critique de la littérature. C'est l'occasion de prendre un peu de recul et de donner les perspectives sur la plateforme d'exécution en temps réel, les stratégies adaptatives, la place du planificateur dans ces stratégies, la notion d'optimisation en temps réel et les qualités des simulateurs en ligne.

### **3.1 Système manufacturier adaptatif intelligent**

L'ajout du noyau d'aide à décision intégré au MES et à l'ERP permet de combler les lacunes des différents systèmes et de proposer un véritable système manufacturier adaptatif intelligent supportant les stratégies adaptatives (Figure 2.1). Le noyau d'aide à la décision couplant des heuristiques à un simulateur apporte les capacités manquantes de modélisation et d'évaluation des décisions. Les capacités de détection sont améliorées par l'intégration des différents systèmes pour permettre de réagir à des perturbations apparaissant à différents niveaux de l'organisation. L'intégration développe aussi les possibilités de propagation des décisions et l'automatisation de certaines actions.

L'ajout d'un serveur d'intégration facilite l'ajout de nouveaux systèmes à la plateforme existante en limitant le nombre d'interfaces à développer. L'utilisation de web services et de message suivant la norme ISA-95 permet une grande réutilisation des interfaces existantes au niveau du serveur d'intégration. Il ne reste au nouvel entrant qu'à avoir connaissance des contrats de service et la correspondance entre les champs des messages et ses données internes. Il faut bien entendu que tous ces systèmes soient en mesure de communiquer par ce biais-là, mais la popularité de cette technologie est telle que la quasi-totalité des systèmes d'information (ERP et MES) en est capable. Seuls les simulateurs que nous avons utilisés nous ont posé problème en la matière. Un des grands intérêts des web services tient dans la généricité des interfaces créées. En effet, il devient possible de changer l'un des systèmes sans pour autant perturber l'architecture. S'il est

peu fréquent de changer un ERP ou un MES, cela peut s'avérer très utile en ce qui concerne le noyau d'optimisation et le simulateur.

### **3.2 Stratégie adaptative**

Comme on a pu le voir avec les articles 2 et 3 sur des cas d'étude, l'adoption d'une stratégie adaptative pour l'ordonnancement améliore les performances du système de production grâce à la réactivité de la nouvelle stratégie. Cette stratégie se déclenche lors de l'occurrence d'événement d'un ou plusieurs types prédéfinis lors de la phase de conception. Il faut toutefois porter une attention particulière sur la fréquence de ces événements et l'ampleur des changements que la stratégie va proposer. En effet, des changements trop fréquents rendront la stratégie de planification instable et enlèveront toute visibilité au personnel de l'entreprise. En dehors d'un impact négatif sur les performances, le risque est de décrédibiliser la stratégie aux yeux du planificateur avec pour résultat son contournement et son abandon progressif.

### **3.3 Optimisation et faisabilité**

Les cédules optimisées ne sont généralement pas très robustes quand elles se retrouvent confrontées au monde réel. Les ateliers de production sont des environnements dynamiques et instables. Au moindre changement, les qualités optimales d'une cédule peuvent rapidement disparaître. De plus, les problèmes qui requièrent un ajustement de la cédule de production sont très souvent urgents. Le problème de réordonnancement doit être résolu en temps réel ou du moins dans un court laps de temps. Cela ne laisse généralement pas le temps aux heuristiques du noyau d'aide à la décision de converger vers la solution optimale. Ce temps doit donc être mis à contribution pour trouver la meilleure solution possible. Plus que la poursuite absolue de l'optimalité, nous cherchons plutôt à maintenir la situation entre certaines limites tout en accomplissant notre objectif.

D'autre part, l'urgence de la situation fait en sorte qu'il n'y a pas de temps à perdre à résoudre les problèmes résultants de la génération de solutions non faisable. Si la solution proposée n'est pas réalisable, le planificateur perdra confiance dans le système et le contournera à la première

occasion. Pour que la solution proposée soit réalisable, le modèle utilisé doit être suffisamment précis et initialisé avec des données reflétant bien la situation actuelle du système de production. Malheureusement, le niveau de précision risque aussi d'avoir un impact sur le temps d'évaluation d'une solution et devra être adapté pour que l'outil d'aide à la décision reste utilisable en conditions réelles.

### **3.4 Place du planificateur dans le processus de décision**

Une des principales raisons à l'origine de la désaffection des planificateurs pour les logiciels d'ordonnancement provient du fait que ces logiciels utilisent souvent des données qui ne sont pas mises à jour assez fréquemment, les capacités de production notamment. Cela oblige alors le planificateur à modifier la solution générée par l'outil pour tenir compte de contraintes inconnues de celui-ci. L'effort peut être alors plus ou moins important. Avec cette plateforme, les données utilisées pour évaluer les décisions proviennent directement du système de production et reflètent son état précis au moment où la stratégie adaptative s'enclenche. La solution générée devrait donc être assez proche d'une solution réellement réalisable et les modifications apportées par le planificateur devraient être minimales. Malgré tout, cette étape reste indispensable, car certaines contraintes restent difficilement modélisables.

Toutefois, bien souvent le planificateur n'intervient dans le processus d'aide à la décision qu'après le traitement du logiciel, pour par exemple modifier une séquence en fonction de ses connaissances de la situation dans l'atelier. L'intervention humaine est là pour corriger les erreurs et pallier les lacunes du noyau d'optimisation. Pourtant, les connaissances du planificateur sont disponibles bien plus tôt et pourraient être utilisées pour initier et guider le processus d'optimisation. En plaçant certaines opérations sur le calendrier ou en bloquant temporairement l'utilisation de certaines ressources sur une période donnée, il ajouterait des contraintes que le noyau d'optimisation devrait prendre en compte et que la solution générée respecterait. Le processus entre le planificateur et le noyau d'optimisation peut même être alors itératif jusqu'à ce que la solution satisfasse le planificateur.

Le processus d'optimisation apporte au planificateur ses capacités de calcul et d'évaluation, mais les deux travaillent de concert pour atteindre les objectifs du planificateur. Pour cela encore faut-il que les deux aient les mêmes objectifs. Bien souvent, les modèles d'ordonnancement et les heuristiques utilisés fonctionnent avec des objectifs financiers qui sont étrangers au planificateur à ce niveau de planification. Son principal objectif est de faire en sorte que les ordres soient produits en temps et en heure, mais il a aussi de nombreux objectifs secondaires (répartition de la charge, utilisation des ressources, etc.) qui feront qu'une séquence sera préférable à une autre. Ces objectifs sont bien souvent antagonistes et varient d'une industrie à l'autre. Pour que le noyau d'aide à la décision épaulé efficacement le planificateur, il doit être fait à sa mesure, tenir compte de ses objectifs et d'évaluer les solutions générées en fonction de ses mesures de performance. Ainsi, le système d'aide à la prise de décision devient un outil d'amélioration des performances du système de production, mais il peut aussi devenir un outil de transfert de connaissance et d'expérience entre les différents planificateurs de l'entreprise.

### 3.5 Qualité d'un simulateur utilisé en ligne

Au court de notre recherche, nous avons été amenés à tester différents simulateurs (3 pour les 3 derniers articles). Sans entrer dans des considérations sur le langage de modélisation utilisé par le logiciel pour générer les modèles, ces expériences nous ont permis de définir les qualités que se doit de posséder un simulateur pour être un bon simulateur en temps réel utilisable en industrie. Les principales qualités sur lesquelles nous aimerions mettre l'accent sont les suivantes :

- La capacité à pouvoir **créer des entités** après la compilation du modèle. Cette capacité peut avoir un grand impact sur le temps de simulation. Ce comportement est permis pour les entités à durée de vie limitée, entrant et sortant du système. Mais cela n'est souvent pas permis pour les autres entités représentant par exemple les ressources de production. Grâce à cette capacité, le modèle gagne en légèreté puisque seules les ressources réellement disponibles sont ajoutées au modèle qui gèrera potentiellement moins d'entités. Il gagne aussi en flexibilité puisque l'ajout de nouvelles ressources d'un type déjà existant ne nécessite pas de recompilation du modèle. Cela permet aussi la

réutilisation des modèles développés pour une usine lors de l'implantation dans une autre usine fonctionnant sur le même principe. Ce dernier point est particulièrement intéressant quand les processus de production sont bien normalisés dans toutes les usines d'un même groupe. Cela peut être mené à bien soit par le chargement d'un fichier texte, soit par des requêtes sur une base de données, soit à la réception de message (formats propriétaires ou web services).

- La capacité à pouvoir **initialiser des entités** après la compilation du modèle. Cette capacité est indispensable pour un fonctionnement en temps réel. Elle permet au simulateur d'initialiser les ressources en fonction de leur état réel dans l'atelier. Là encore, cela peut se faire soit par le chargement d'un fichier texte, soit par des requêtes sur une base de données, soit à la réception de message (formats propriétaires ou web services). Si le simulateur le permet, il est possible de combiner la création et l'initialisation des entités.
- La capacité de pouvoir **lire et écrire dans une base de données**. Cette capacité n'est pas foncièrement indispensable, mais elle est d'une aide précieuse dans bien des cas. Par exemple, elle simplifie grandement l'implantation d'une mémoire telle que proposée dans notre quatrième article (Annexe 4). D'un point de vue plus pratique dans un cadre de recherche, cela permet de centraliser de nombreux résultats et indicateurs et de faciliter leur analyse.
- La capacité de **se comporter comme un web service et d'effectuer des requêtes à un web serveur**. L'intérêt majeur est ici de faciliter l'intégration du simulateur dans l'architecture d'exécution. En se comportant comme un web service, l'évaluation d'une séquence se limite alors à une requête au web serveur. Le simulateur devient ainsi interchangeable si nécessaire et pourrait même être remplacé par d'autres méthodes si nécessaire. En étant capable de faire des requêtes à un web serveur, le simulateur devient proactif, il est en mesure de demander de manière synchrone au serveur d'intégration l'état des ressources avant que la simulation ne débute. Si en plus le simulateur a la



capacité de lire et d'écrire dans une base de données, cette requête peut être effectuée une fois au début de la stratégie adaptative. Les états initiaux sont alors stockés dans la base de données et réutilisables à volonté.

- La capacité d'**exploiter au maximum les ressources matérielles** de l'ordinateur sur lequel le simulateur fonctionne. Ces dernières années, plusieurs avancées majeures ont été faites dans le domaine des processeurs. Le mode d'adressage est passé de 32 à 64 bits, permettant à nos ordinateurs d'exploiter plus de mémoire vive. Mais surtout, les processeurs sont maintenant multicœurs. Bien exploités, ces cœurs augmentent la vitesse de calcul. L'exploitation des multicœurs par les simulateurs commerciaux n'en est qu'à ses débuts et à cette heure un seul (en version non finalisée) en tire explicitement profit. Une autre possibilité consiste à reporter les calculs lourds sur le processeur des cartes graphiques (GPU, *Graphical Processor Unit*). NVIDIA propose déjà un kit de développement pour les applications écrites en langage C et a annoncé son intention de créer un compilateur frontal aux compilateurs Fortran qui serait disponible à partir du mois de novembre 2009. Cette technologie permet des accélérations spectaculaires des performances comme cela a été observé dans les simulateurs testant le pliage des protéines.

Ces trois derniers points sont aussi valables pour le moteur d'optimisation intégrant les heuristiques.

### 3.6 Opportunités de recherche

En dépit de toutes les avancées dans le domaine du contrôle de la production, nous avons remarqué qu'aussi bien la communauté industrielle que la communauté académique se sont principalement attaquées au problème de production adaptative du point de vue de l'intégration des systèmes d'information. Alors que les plus récents systèmes d'exécution sont en mesure de détecter les événements aussi bien dans les processus d'affaires que dans les processus de production, de connecter les ateliers, les entreprises et la chaîne logistique élargie et de fournir une régulation des réponses, ils ne fournissent toujours pas d'alternatives pour résoudre les

problèmes en temps réel. En effet, les stratégies de production adaptative doivent combiner les processus d'affaires, les outils, les logiciels et les technologies d'optimisation en ligne afin de faire en sorte que les bonnes ressources collaborent au bon moment pour résoudre les exceptions. Les domaines de recherche potentiels sont nombreux, allant du développement de nouveaux algorithmes de réordonnancement au développement d'indicateurs de performance. Si l'on veut que les industriels fassent le saut vers le concept d'adaptivité, le monde de la recherche doit prendre en compte les contraintes spécifiques des ateliers de production dynamiques, le grand nombre de variables entrant en jeu dans les modèles, le temps de résolution restreint et reconnaître les spécificités de chaque source de données.

La combinaison de la simulation en ligne et des algorithmes d'optimisation est maintenant bien établie dans le monde académique, mais c'est loin d'être le cas dans le monde industriel. Comme nous l'avons montré, intégrée à un MES et à un ERP, la simulation est un puissant outil pour évaluer différents scénarios.

L'analyse de sensibilité sur la base de scénarios est extrêmement importante à un horizon de prise de décision à court terme. À ce niveau, le système d'aide à la décision est très utile en raison de l'importante quantité de données à traiter et de l'impact immédiat des mesures décidées et à mettre en œuvre.

### **3.7 Conclusion**

Dans ce chapitre, nous avons discuté des aspects techniques et conceptuels liés à notre recherche. Nous en concluons de l'intérêt des stratégies adaptatives d'ordonnancement dans un contexte industriel. Du point de vue de l'intégration des systèmes, l'ajout d'un serveur d'intégration et l'utilisation de web services ajoutent de la flexibilité à l'ensemble de la plateforme d'exécution en temps réel. Nous mettons aussi en garde contre une possible instabilité des décisions si la stratégie adaptative est mal contrôlée. Nous limitons aussi le sens du concept d'optimisation quand il est associé à la notion de temps réel. Nous insistons sur la place centrale du planificateur dans le processus d'aide à la décision. Et finalement, nous esquissons le portrait du simulateur en ligne idéal pour une utilisation industrielle.

## CONCLUSION

Une stratégie adaptative se définit comme une stratégie réagissant à un changement d'état de l'environnement à contrôler. Par sa compréhension de cet environnement et sa connaissance des ressources à sa disposition, elle permet de réagir aux changements qui se profilent en faisant collaborer les différents acteurs et systèmes de l'environnement.

L'automatisation croissante des ateliers de production, la démocratisation des progiciels de gestion intégrée (ERP) et l'essor des systèmes d'exécution manufacturière font en sorte que les entreprises modernes ont accès à de plus en plus de données. Malheureusement, une intégration déficiente de ces systèmes et des lacunes fonctionnelles font que ces données restent inexploitées. Les ERP sont déconnectés de l'atelier et sa méthode de planification hérite des défauts de la méthodologie MRPII (capacité finie et déterminisme). Les MES sont le lien entre l'atelier et le reste de l'entreprise et leur capacité à traiter l'information en temps réel est précieuse. Tout est là pour pouvoir mettre en oeuvre une stratégie adaptative de planification, mais l'approche d'optimisation en temps réel est toujours manquante. En recherche opérationnelle, si des approches existent pour résoudre ces problèmes, elles sont généralement appliquées hors ligne et hors de toute connexion avec l'environnement de production. Cela a pour conséquence une mauvaise évaluation des contraintes et des défis d'une utilisation en temps réel sont, sans parler de la génération de solution en décalage avec la réalité du système de production.

Notre objectif principal était de proposer un modèle intégré de planification et d'ordonnancement réactif et d'opérer ainsi une jonction entre les domaines de la recherche opérationnelle et des systèmes d'information. Plus concrètement, le modèle vise l'implantation de composantes d'aide à la prise de décision en temps réel ainsi que des mécanismes de boucle rétroactive intégrant l'optimisation et les techniques de simulation au sein d'applications ERP et MES permettant ainsi de connecter l'atelier de production au reste de l'entreprise. La plateforme ainsi mise en place permet de répondre en temps réel aux divers aléas survenant dans l'atelier et peut être étendue au-delà de la problématique de l'ordonnancement. Pour cela, nous avons commencé par montrer qu'en l'état actuel, l'intégration des MES et des ERP est insuffisante pour combler la nécessité

d'une optimisation en temps réel, mais qu'elle est une étape nécessaire à la mise en place d'une stratégie adaptative de planification. Nous avons par la suite proposé une plateforme d'exécution en temps réel et nous avons démontré l'apport déterminant de cette plateforme et des stratégies adaptatives sur les performances des ateliers de production. Une première étude portait sur un scénario de traitement des ordres de modification technique (ECO) dans l'industrie aéronautique. L'objectif était à la fois de minimiser la durée de la cédule de production tout en faisant en sorte que l'ECO soit produit rapidement. Une seconde étude se basait sur un scénario d'ordonnancement dans l'industrie de l'aluminium ayant pour objectif de minimiser le déclassement de l'aluminium de bonne qualité pour produire des alliages de moins bonne qualité. Nous avons aussi proposé des mécanismes qui permettent sous certaines hypothèses de réduire le temps de calcul du processus d'optimisation en temps réel quand la procédure d'évaluation des solutions générées est longue. Ces mécanismes ont été proposés pour les algorithmes génétiques, mais peuvent être étendus à toutes les heuristiques utilisant une procédure d'évaluation.

Dans l'article intitulé « *Manufacturing execution system – a literature review* », nous faisons une revue de la littérature des MES. Ce système est la clé de la communication entre les systèmes d'information et les ateliers de production. Cet article est une exploration préliminaire à la proposition d'une architecture systémique. Il nous permet de valider que l'intégration des MES et des ERP n'est pas suffisante pour supporter les stratégies adaptatives de planification et qu'un module d'aide à la décision doit être ajouté.

Dans l'article intitulé « *Engineering Change Order Processing in ERP Systems: An Integrated Reactive Model* », nous proposons un modèle d'ordonnancement réactif dans lequel un ERP, un MES et un noyau d'aide à la décision sont intégrés. La plateforme d'exécution est mise dans un contexte de traitement des ordres de changement technique dans l'industrie aéronautique. Le noyau d'aide à la décision se compose d'un algorithme génétique et d'un simulateur. Ce cas d'étude nous permet de montrer que la plateforme proposée supporte efficacement les stratégies adaptatives et que ces stratégies apportent un bénéfice aux performances du système de production.

Dans l'article intitulé « *Development and integration of a reactive real-time decision support system in the aluminum industry* », la plateforme proposée dans l'article précédent est enrichie d'un serveur d'intégration. La flexibilité et la robustesse de la plateforme sont ainsi améliorées. Cette fois-ci, le contexte étudié est celui de la planification dans l'industrie de l'aluminium. Le noyau d'aide à la décision se compose toujours d'un algorithme génétique et d'un simulateur, mais le modèle de simulation est bien plus important que précédemment et la stratégie est mise en œuvre plus fréquemment. Ce cas d'étude nous a permis de valider le bénéfice des stratégies adaptatives, mais nous a aussi permis de mettre au jour des contraintes liées au noyau d'aide à la décision et à son utilisation en temps réel.

Dans l'article intitulé « *Enhancing Genetic Algorithm with predator mechanisms for On-line Optimization* », nous nous plaçons dans le contexte particulier d'utilisation en temps réel d'un algorithme génétique ayant une procédure d'évaluation longue et nous proposons deux mécanismes pour le réduire sans détériorer les qualités de convergence de l'heuristique. L'utilisation d'une mémoire intergénérationnelle et d'un mécanisme de relaxation de l'évaluation, aussi appelé prédateur, nous a permis à la fois de réduire le temps de calcul, mais aussi d'améliorer les performances de la solution. Ces mécanismes peuvent être étendus à d'autres heuristiques sous certaines conditions que nous avons formulées.

La nature de plus en plus complexe et stochastique des ateliers de production est un obstacle à l'application des méthodes traditionnelles de planification. La déconnexion des outils d'ordonnancement rend les solutions qu'ils génèrent inapplicables en l'état et demande au planificateur un important effort de mise à jour manuelle de cédule de production. L'utilisation conjointe de la plateforme d'exécution en temps réel et des stratégies adaptatives permet de réagir aux événements imprévus surgissant dans l'atelier ou ailleurs dans l'entreprise. Elle permet aussi d'évaluer les solutions générées par le noyau d'aide à la décision en utilisant l'état réel des ressources. Une fois les décisions prises par le planificateur, celles-ci sont alors automatiquement propagées au sein de la plateforme d'exécution.

L'utilisation des données de production en temps réel permet de rétablir le dialogue entre le planificateur et l'outil d'ordonnancement. La confiance est rétablie, mais peut être rapidement perdue. En effet si la fréquence et l'amplitude des changements dans les décisions (principalement à court terme) sont trop importantes, cette instabilité sera mal perçue par les utilisateurs qui remettront en cause l'utilité de l'outil. De plus, pour que l'utilité de l'outil soit pleine et entière, il doit calquer sa mesure des performances sur les indicateurs qu'utilise le planificateur pour prendre ses décisions.

Du côté de la recherche opérationnelle, le contexte d'utilisation en temps réel est un défi. Le temps imparti à la résolution n'est bien souvent pas suffisant pour trouver la solution optimale au problème. Les heuristiques sont donc utilisées dans l'optique de trouver la meilleure solution possible dans un temps limité. De plus, les solutions proposées doivent être réalisables pour minimiser le temps que le planificateur passera à modifier ces solutions.

À l'heure actuelle, les connaissances du planificateur sont utilisées *a posteriori*. Or, ces connaissances, le planificateur les possède dès le déclenchement de la stratégie adaptative. Il serait alors intéressant que le dialogue entre le noyau d'aide à la décision et le planificateur s'établisse bien plus tôt et ainsi permettre à l'utilisateur d'initier et de guider le processus d'optimisation en lui imposant des contraintes supplémentaires. Un processus itératif de modifications successives de la cédule par l'homme et par la machine aboutirait à une solution plus satisfaisante pour le planificateur et le processus d'optimisation serait plus transparent puisque l'utilisateur serait plus impliqué. Pour se faire, les avancées dans le domaine des systèmes mixtes seraient à explorer.

## BIBLIOGRAPHIE

- Adshead, A. (2007). From shopfloor to top floor manufacturing execution systems. *Manufacturing Computer Solutions*, 13(2), 44-45.
- Akkermans, H. A., Bogerd, P., Yucesan, E., & Van Wassenhove, L. N. (2003). The impact of ERP on supply chain management: Exploratory findings from a European Delphi study. *European Journal of Operational Research*, 146(2), 284-301.
- Alberts, D. S., & Hayes, R. E. (2003). *Power to the Edge: Command and Control in the Information Age*: DoD Command and Control Research Program.
- Aytug, H., Bhattacharyya, S., Koehler, G. J., & Snowdon, J. L. (1994). A review of machine learning in scheduling. *IEEE Transactions on Engineering Management*, 41(2), 165-171.
- Azadivar, F., & Lee, Y.-H. (1988). Optimization of discrete variable stochastic systems by computer simulation. *Mathematics and Computers in Simulation*, 30(4), 331-345.
- Azadivar, F., & Tompkins, G. (1999). Simulation optimization with qualitative variables and structural model changes: A genetic algorithm approach. *European Journal of Operational Research*, 113(1), 169-182.
- Baliga, J. (1998). MES and CIM: at the center of productivity. *Semiconductor International*, 21(8), 104-106, 108, 110, 112.
- Boesel, J., Bowden Jr, R. O., Glover, F., Kelly, J. P., & Westwig, E. (2001). Future of simulation optimization. *2001 Winter Simulation Conference, Arlington, VA* (Vol. 2, pp. 1466-1469) Institute of Electrical and Electronics Engineers Inc.
- Carson, Y., & Maria, A. (1997). Simulation optimization: Methods and applications. *1997 Winter Simulation Conference, Atlanta, GA* (pp. 118-126) IEEE Piscataway NJ USA.
- Chan, F. T. S., & Chan, H. K. (2004). A comprehensive survey and future trend of simulation study on FMS scheduling. *Journal of Intelligent Manufacturing*, 15(1), 87-102.

Charles, A., Lauras, M., & Van Wassenhove, L. (2009). Towards an agile supply chain: lessons learned from humanitarians. *International Conference on Industrial Engineering and Systems Management, Montréal (QC) Canada*.

Chaudhry, S. S., & Luo, W. (2005). Application of genetic algorithms in production and operations management: A review. *International Journal of Production Research*, 43(19), 4083-4101.

Chazalet, A., & Lalanda, P. (2007). Deployment of services-oriented applications integrating physical and IT systems. *21st International Conference on Advanced Information Networking and Applications, AINA 2007, Niagara Falls, ON, Canada* (pp. 38-45) Institute of Electrical and Electronics Engineers Inc. Piscataway NJ 08855-1331 United States.

Chen, K.-Y., & Wu, T.-C. (2005). Data warehouse design for manufacturing execution systems. *2005 IEEE International Conference on Mechatronics, ICM '05, Taipei, Taiwan* (Vol. 2005, pp. 751-756) Institute of Electrical and Electronics Engineers Computer Society Piscataway NJ 08855-1331 United States.

Cheng, F.-T., Lin, M.-T., & Lee, R.-S. (2000). Developing a web-enabled equipment driver for semiconductor equipment communications. *ICRA 2000: IEEE International Conference on Robotics and Automation, San Francisco, CA* (Vol. 3, pp. 2203-2210) Institute of Electrical and Electronics Engineers Inc. Piscataway NJ USA.

Cheng, F.-T., Yang, H.-C., Kuo, T.-L., Feng, C., & Jeng, M. (2000). Modeling and analysis of equipment managers in manufacturing execution systems for semiconductor packaging. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 30(5), 772-782.

Choi, S. H., & Kang, S. G. (2005). An integrated platform for distributed knowledge-based production scheduling. *6th International Conference on Computational Intelligence and Multimedia Applications, ICCIMA 2005, Las Vegas, NV, United States* (pp. 111-117) Institute of Electrical and Electronics Engineers Computer Society Piscataway NJ 08855-1331 United States.



- Chong, P., & Chen, F. F. (1998). Real-time control and scheduling of flexible manufacturing systems: an ordinal optimisation based approach. *International Journal of Advanced Manufacturing Technology*, 14(10), 775-786.
- Christopher, M., & Towill, D. (2000). Supply chain migration from lean and functional to agile and customized. *Supply Chain Management: an International Journal*, 5(4), 206-213.
- Dangelmaier, W., Mahajan, K. R., Seeger, T., Klopper, B., & Aufenanger, M. (2006). Simulation assisted optimization and real-time control aspects of flexible production systems subject to disturbances. *2006 Winter Simulation Conference, Monterey, CA* (pp. 1785-1795)IEEE.
- Di Marzo, S. G., Karageorgos, A., Rana, O. F., & Zambonelli, F. (2004). Engineering self-organising systems: nature-inspired approaches to software engineering. *Lecture Notes in Artificial Intelligence, Verlag Heidelberg D-69121 Germany* (Vol. 2977, pp. 299)Springer.
- Diep, D., Massotte, P., & Meimouni, A. (2003). A distributed manufacturing execution system implemented with agents: the PABADIS model. *IEEE International Conference on Industrial Informatics, Banff, Alta., Canada* (pp. 301-306)IEEE.
- Dilts, D. M., Boyd, N. P., & Whorms, H. H. (1991). The evolution of control architectures for automated manufacturing systems. *Journal of Manufacturing Systems*, 10(1), 79-93.
- ElMaraghy, H. A., Abdallah, I. B., & ElMaraghy, W. H. (1998). On-line simulation and control in manufacturing systems. *CIRP Annals - Manufacturing Technology*, 47(1), 401-404.
- Erl, T. (2005). *Service-Oriented Architecture: Concepts, Technology, and Design*. Upper Saddle River: Prentice Hall PTR.
- Gebus, S. (2006). *Knowledge-based decision support systems for production optimization and quality improvement in the electronics industry*. D.Sc. inédit, Oulun Yliopisto (Finland)
- Georges, J.-P., Krommenacker, N., Divoux, T., & Rondeau, E. (2006). A design process of switched Ethernet architectures according to real-time application constraints. *Engineering Applications of Artificial Intelligence*, 19(3), 335-344.

- Goldman, S. L., & Nagel, R. N. (1993). Management, technology and agility. The emergence of a new era in manufacturing. *International Journal of Technology Management*, 8(1-2), 18.
- Goldman, S. L., Nagel, R. N., & Preiss, K. (1995). Agile competitors and virtual organizations. *Manufacturing Review*, 8(1), 59-67.
- Gonzalez, A. J., & Dankel, D. D. (1993). *The Engineering of Knowledge-Based Systems*. Englewood Cliffs, NJ: Prentice Hall.
- Gunasekaran, A. (1998). Agile manufacturing: Enablers and an implementation framework. *International Journal of Production Research*, 36(5), 1223-1247.
- Harhalakis, G., Lin, C.-P., Mark, L., & Muro-Medrano, P. (1994). Implementation of rule-based information systems for integrated manufacturing. *IEEE Transactions on Knowledge and Data Engineering*, 6(6), 892-908.
- Harhalakis, G., Lin, C.-P., Mark, L., & Muro-Medrano, P. (1995). Structured representation of rule-based specifications in CIM using updated petri nets. *IEEE Transactions on Systems, Man and Cybernetics*, 25(1), 130-144.
- Holst, L. (2001). *Integrating Discrete-Event Simulation into the Manufacturing System Development Process*. Lund University, Department of Mechanical Engineering, Division of Robotics, Lund, Sweden.
- Hopp, W., & Spearman, M. (2000). *Factory Physics*. New York , NY: McGraw-Hill.
- Hormozi, A. H. (2001). Agile manufacturing: the next logical step. *Benchmarking: An International Journal*, 8(2), 132-143.
- Howells, R. (2000). ERP needs shop-floor data. *Manufacturing Engineering*, 125(4), 54-62.
- Huang, C.-Y. (2002). Distributed manufacturing execution systems: A workflow perspective. *Journal of Intelligent Manufacturing*, 13(6), 485-497.
- Instrument Society of America. (1995). *ANSI/ISA-88.01-1995, Batch Control – Part 1: Models and Terminology*, 978-1-55617-745-3.

Instrument Society of America. (2000). *ANSI/ISA-95.00.01-2000: Enterprise Control System Integration*, 978-1-55617-727-9.

Instrument Society of America. (2007). *ANSI/ISA-95.00.05-2007: Integration, Part 5: Business-to-Manufacturing Transactions*, 978097923-4354.

Instrument Society of America. (2009). *ISA draft 95.00.06: Integration, Part 6: Manufacturing Operations Management Transactions*, draft.

Ip, W. H., Kwong, C. K., & Fung, R. (2000). Design of maintenance system in MRPII. *Journal of Quality in Maintenance Engineering*, 6(3), 177-191.

Ismail, H. S., Snowden, S. P., Poolton, J., Reid, I. R., & Arokiam, I. C. (2006). Agile manufacturing framework and practice. *International Journal of Agile Systems and Management*, 1(1), 11-28.

Jiang, P. Y., Zhou, G., Zhao, G., Zhang, Y. F., & Sun, H. B. (2007). e<sup>2</sup>-MES: an e-service-driven networked manufacturing platform for extended enterprises. *International Journal of Computer Integrated Manufacturing*, 20(2-3), 127-142.

Jones, A. T., Reeker, L. H., & Deshmukh, A. V. (2002). On Information and Performance of Complex Manufacturing Systems. Consulté le 25 juin 2009, tiré de -, [http://www.ifm.eng.cam.ac.uk/mcn/pdf\\_files/part6\\_1.pdf](http://www.ifm.eng.cam.ac.uk/mcn/pdf_files/part6_1.pdf).

Kidd, P. T. (1996). Agile manufacturing: A strategy for the 21st century. *IEE Colloquium on Agile Manufacturing, London, UK* (pp. 6)IEE Stevenage Engl.

Koch, C. (2001). Why your integration efforts end up looking like this... *CIO Magazine*, 15, 98-108.

Koren, Y. (2003). Reconfigurable manufacturing systems. 42(7), 572-582.

Lee, Y.-H., & Iwata, K. (1991). Part ordering through simulation-optimization in an FMS. *International Journal of Production Research*, 29(7), 1309-1323.

- Leitão, P., & Restivo, F. (1999). A layered approach to distributed manufacturing. *ASI'99 International Conference, Leuven, Belgium* (pp. 21-23).
- Li, S., Wu, Z., & Pang, X. (2005). Job shop scheduling in real-time cases. *International Journal of Advanced Manufacturing Technology*, 26(7-8), 870-875.
- Lin, C.-P., Jeng, L.-D., Lin, Y.-P., & Jeng, M. (2005). Management and control of information flow in CIM systems using UML and Petri nets. *International Journal of Computer Integrated Manufacturing*, 18(2-3), 107-121.
- Lin, C.-P., & Jeng, M. (2006). An expanded SEMATECH CIM framework for heterogeneous applications integration. *IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans*, 36(1), 76-90.
- Linthicum, D. S. (1999). *Enterprise Application Integration*. MA: Addison-Wesley.
- Liu, W., Chua, T. J., Lam, J., Wang, F. Y., Cai, T. X., & Yin, X. F. (2002). APS, ERP and MES systems integration for Semiconductor Backend Assembly. *7th International Conference on Control, Automation, Robotics and Vision, ICARCV 2002, Marine Mandarin, Singapore* (Vol. 1, pp. 1403-1408) Nanyang Technological University Singapore, Singapore.
- Mandal, P., & Gunasekaran, A. (2002). Application of SAP R/3 in on-line inventory control. *International Journal of Production Economics*, 75(1-2), 47-55.
- Martin, R. (2004). Produce to Demand Is a Manufacturing Strategy, Not an MES Project. Consulté le <http://www.amrresearch.com/Content/View.asp?pmillid=17201>.
- McKay, K. N., Safayeni, F. R., & Buzacott, J. A. (1988). Job-shop scheduling theory: what is relevant? , 18(4), 84-90.
- Mejia, G., & Odrey, N. G. (2005). An approach using petri nets and improved heuristic search for manufacturing system scheduling. *Journal of Manufacturing Systems*, 24(2), 79-92.
- Mertens, P., Falk, J., & Spieck, S. (1994). Comparisons of agent approaches with centralized alternatives based on logistical scenarios. *Information Systems*, 19(8), 699-709.

MESA. (1997a). White Paper #02: MES Functionalities and MRP to MES Data Flow Possibilities. Consulté le 24 juin 2009, tiré de -, <http://www.mesa.org/knowledge-base/details.php?id=49>.

MESA. (1997b). White Paper #06: MES Explained: A High Level Vision for Executives. Consulté le 24 juin 2009, tiré de -, <http://www.mesa.org/knowledge-base/details.php?id=53>.

MESA. (2000). White Paper #03: The Controls Layer: Controls Definition and MES to Controls Data Flow Possibilities. Consulté le 24 juin 2009, tiré de -, <http://www.mesa.org/knowledge-base/details.php?id=50>.

Moon, Y. B., & Phatak, D. (2005). Enhancing ERP system's functionality with discrete event simulation. *Industrial Management and Data Systems*, 105(9), 1206-1224.

Mustafa, Y., & Mejabi, O. (1999). An approach for developing flexible MRP systems. *Information Systems Management*, 16(2), 58-63.

Nikolopoulos, K., Metaxiotis, K., Lekatis, N., & Assimakopoulos, V. (2003). Integrating industrial maintenance strategy into ERP. *Industrial Management and Data Systems*, 103(3-4), 184-191.

Ogbu, F. A., & Smith, D. K. (1990). Application of the simulated annealing algorithm to the solution of the  $n/m/C_{\max}$  flowshop problem. *Computers & Operations Research*, 17(3), 243-253.

Owen, T., & Parker, K. (1999). One app's ceiling is another app's floor. *Manufacturing Systems*, 17(10), 6 pp.

PABADIS Group. (2001). Revolutionizing Plant Automation : the PABADIS Approach. Consulté le -, [http://www.pabadis.org/downloads/pabadis\\_del\\_6\\_3.pdf](http://www.pabadis.org/downloads/pabadis_del_6_3.pdf).

Qiu, R. G. (2002). A data fusion framework for an integrated plant-wide information system. *5th International Conference on Information Fusion, Annapolis, MD* (Vol. 1, pp. 101-107 vol.101)Int. Soc. Inf. Fusion.

- Qiu, R. G. (2003). E-manufacturing: The keystone of a plant-wide real time information system. *Journal of the Chinese Institute of Industrial Engineers*, 20(3), 266-274.
- Qiu, R. G., Burda, R., & Chylak, R. (2002). Distributed WIP control in advanced semiconductor manufacturing. *13th Annual IEEE/SEMI Advanced Semiconductor Manufacturing Conference, Boston, MA, United States* (pp. 49-54)IEEE Inc.
- Reyes, A., Yu, H., Kelleher, G., & Lloyd, S. (2002). Integrating Petri nets and hybrid heuristic search for the scheduling of FMS. *Computers in Industry*, 47(1), 123-138.
- Rizzi, A., & Zamboni, R. (1999). Efficiency improvement in manual warehouses through ERP systems implementation and redesign of the logistics processes. *Logistics Information Management*, 12(5), 367-377.
- Rockwell. (2006). Making sense of e-manufacturing: A roadmap for manufacturers. Consulté le 25 juin 2009, tiré de -, <http://www.automation.rockwell.com>.
- Rondeau, P. J., & Litteral, L. A. (2001). Evolution of manufacturing planning and control systems: From reorder point to enterprise resource planning. *Production and Inventory Management Journal*, 42(2), 1-7.
- Ruiz, R., Carlos Garcia-Diaz, J., & Maroto, C. (2007). Considering scheduling and preventive maintenance in the flowshop sequencing problem. *Computers and Operations Research*, 34(11), 3314-3330.
- Saad, S. M., & Gindy, N. N. (1998). Handling internal and external disturbances in responsive manufacturing environments. *Production Planning and Control*, 9(8), 760-770.
- Salam, A. F., Rao, H. R., & Bhattacharjee, S. (1999). Internet-based technologies: value creation for the customer and the value chain across industries. *AIS AMCIS 99: 1999 Americas Conference on Information Systems, Milwaukee, WI* (pp. 538-540)Assoc. Inf. Syst.
- Sanchez, L. M., & Nagi, R. (2001). A review of agile manufacturing systems. *International Journal of Production Research*, 39(16), 3561-3600.

- SEMATECH Inc. (1998). *Computer Integrated Manufacturing Framework Specification Version 2.0*. Austin, TX.
- Sharifi, H., & Zhang, Z. (2001). Agile manufacturing in practice. Application of a methodology. *International Journal of Operations and Production Management*, 21(5/6), 772-794.
- Shen, W., & Norrie, D. H. (1999). Agent-based systems for intelligent manufacturing: a state-of-the-art survey. *Knowledge and Information Systems*, 1(2), 129-156.
- Sherehiy, B., Karwowski, W., & Layer, J. K. (2007). A review of enterprise agility: Concepts, frameworks, and attributes. *International Journal of Industrial Ergonomics*, 37(5), 445-460.
- Sheridan, J. H. (1993). Agile manufacturing. Beyond lean production. *Industry Week*, 242(8), 30.
- Shukla, C. S., & Chen, F. F. (1996). State of the art in intelligent real-time FMS control: A comprehensive survey. *Journal of Intelligent Manufacturing*, 7(6), 441-455.
- Simao, J. M., Stadzisz, P. C., & Morel, G. (2006). Manufacturing execution systems for customized production. *Journal of Materials Processing Technology*, 179(1-3), 268-275.
- Slater, D. (2000). Talk to Your Plants. Consulté le 25 juin 2009, tiré de -, [http://www.itworld.com/App/670/CIO031500\\_plants\\_content/](http://www.itworld.com/App/670/CIO031500_plants_content/).
- Stadtler, H. (2005). Supply chain management and advanced planning - Basics, overview and challenges. *European Journal of Operational Research*, 163(3), 575-588.
- Stadtler, H., & Kilger, C. (2000). *Supply Chain Management and Advanced Planning - Concepts, Models, Software and Case Studies*. Berlin: Springer.
- Swafford, P. M., Ghosh, S., & Murthy, N. (2006). The antecedents of supply chain agility of a firm: Scale development and model testing. 24(2), 170-188.
- Tang, L., Liu, W., & Liu, J. (2005). A neural network model and algorithm for the hybrid flow shop scheduling problem in a dynamic environment. *Journal of Intelligent Manufacturing*, 16(3), 361-370.

- Tang, Y., & Qiu, R. G. (2004). Integrated design approach for virtual production line-based reconfigurable manufacturing systems. *International Journal of Production Research*, 42(18), 3803-3822.
- Tsourveloudis, N. C., & Valavanis, K. P. (2002). On the measurement of enterprise agility. *Journal of Intelligent and Robotic Systems: Theory and Applications*, 33(3), 329-342.
- Upton, D. M., & McAfee, A. P. (2000). A path-based approach to information technology in manufacturing. *International Journal of Technology Management*, 20(3-4), 354-372.
- Valckenaers, P., Saint Germain, B., Verstraete, P., Hadeli, Zamfirescu, C., & Van Brussel, H. (2004). Ant colony engineering in coordination and control: how to engineer a short term forecasting system. *5<sup>th</sup> International Workshop on Emergent Synthesis, IWES'04, Budapest, Hungary* (pp. 125-132).
- Valckenaers, P., Van Brussel, H., Bonneville, F., Bongaerts, L., & Wyns, J. (1994). IMS test case 5. Holonic manufacturing systems. *IFAC Workshop on Intelligent Manufacturing Systems (IMS'94), Vienna, Aust* (pp. 31-36)Pergamon Press Ltd.
- Valckenaers, P., Van Brussel, H., Hadeli, Bochmann, O., Germain, B. S., & Zamfirescu, C. (2003). On the design of emergent systems: An investigation of integration and interoperability issues. *Engineering Applications of Artificial Intelligence*, 16(4), 377-393.
- Van Dyke, H. (1998). What can agents do in industry, and why? An overview of industrially-oriented R&D at CEC. *Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science)*, 1435(1), 1-18.
- Vieira, G. E., Herrmann, J. W., & Lin, E. (2003). Rescheduling manufacturing systems: A framework of strategies, policies, and methods. *Journal of Scheduling*, 6(1), 39-62.
- Vokurka, R. J., & Fliedner, G. (1998). Journey toward agility. *Industrial Management and Data Systems*, 98(3-4), 165-171.
- Waldrop, M. M. (1993). *Complexity: the emerging science at the edge of order and chaos*. New-York (NY): Touchstone.



Westerlund, T. (1996). ERP and MES integration: Reducing cycle time. *1996 Industrial Computing Conference, ICS/96, Chicago, IL* (Vol. 6, pp. 91-96)Instrument Society of America Research Triangle Park NC USA.

Wildeman, R. C., Leaver, S. C., & Sedov, V. (2008). Market Overview: Manufacturing Execution Systems. Consulté le 3 mars 2009, tiré de <http://www.forrester.com/Research/Document/-Excerpt/0,7211,45736,00.html>.

Wong, D., Paciorek, N., & Moore, D. (1999). Java-based mobile agents. *Communications of the ACM*, 42(3), 92-102.

Wong, K. (2006). We gather today to join ERP and PLM: Marrying enterprise data to product data. *Cadalyst*, 23(9), 42-44.

Wrembel, R., & Koncilia, C. (2006). *Data Warehouses And Olap: Concepts, Architectures And Solutions*. Hershey, PA: IRM Press.

XML Core Working Group. (2003). Extensible Markup Language (XML). Consulté le 2007, tiré de -, <http://www.w3.org/XML/>.

XSL Working Group. (2001). The Extensible Stylesheet Language Family (XSL). Consulté le -, <http://www.w3.org/Style/XSL/>.

Xu, Q., & Qiu, R. G. (2004). Heterogeneous knowledge syntheses for manufacturing information systems. *Jisuanji Jicheng Zhizao Xitong/Computer Integrated Manufacturing Systems, CIMS*, 10(7), 732-736.

Yang, H.-C., Cheng, F.-T., & Huang, D. (1999). Development of a generic equipment manager for semiconductor manufacturing. *1999 7th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA'99), Barcelona, Spain* (Vol. 1, pp. 727-732)IEEE Inc. Piscataway NJ USA.

- Ye, S. X., & Qiu, R. G. (2003). An architecture of configurable equipment connectivity in a future manufacturing information system. *2003 IEEE International Symposium on Computational Intelligence in Robotics and Automation, Kobe, Japan* (Vol. 3, pp. 1144-1149)IEEE.
- Yoon, H. J., & Shen, W. (2006). Simulation-based real-time decision making for manufacturing automation systems: A review. *International Journal of Manufacturing Technology and Management*, 8(1-3), 188-202.
- Yusuf, Y. Y., & Adeleye, E. O. (2002). A comparative study of lean and agile manufacturing with a related survey of current practices in the UK. *International Journal of Production Research*, 40(17), 4545-4562.
- Yusuf, Y. Y., Sarhadi, M., & Gunasekaran, A. (1999). Agile manufacturing: the drivers, concepts and attributes. *International Journal of Production Economics*, 62(1-2), 33-43.
- Zhang, Z., & Sharifi, H. (2000). A methodology for achieving agility in manufacturing organisations. *International Journal of Operations and Production Management*, 20(4), 496-512.
- Zurawski, R., & Zhou, M. (1994). Petri nets and industrial applications: A tutorial. *IEEE Transactions on Industrial Electronics*, 41(6), 567-583.

## **ANNEXE A: MANUFACTURING EXECUTION SYSTEM – A LITERATURE REVIEW**

Référence : Saenz de Ugarte, B., Artiba, A. and Pellerin, R.(2009). Manufacturing execution system - a literature review, *Production Planning & Control*, 20(6), 525- 539

**Abstract** – Since their beginning in the mid 1990s, Manufacturing Execution Systems (MES) have significantly evolved into more powerful and more integrated software applications as computing technologies have advanced. Their functionality coverage has changed significantly and can now provide a common and single system to support most of the manufacturing execution processes from the production order release to the delivery of finished goods. However, MES applications still fall short of adding capabilities for decision-making tools in very dynamic organizations where adaptive execution strategies are required to replenish the supply chain while dynamically responding to unpredicted change. This paper intends to describe what MES have become, present their relationships with other enterprise information systems, and to identify major issues related to MES. We also discuss research areas that need to be explored in order to resolve the increased complexity of execution systems and to fulfil the continuing customer needs for faster real-time response and expanded functionality coverage.

**Keywords** – Manufacturing; execution system; MES; ERP; adaptive manufacturing.

## **A.1 Introduction**

For more than 25 years, companies have invested in information systems to achieve productivity gains that have caused the enterprise information system market to grow steadily. Some information systems, such as the Enterprise Resource Planning systems (ERP), gained a major status in the market, as 80% of the Fortune 500 firms are now using ERP to manage their operations and a growing number of SMEs are adopting the same strategy. But for most of this period, the information system specialists have not paid attention to the shop floor (Holst, 2001). In that context, it is not surprising to note that production departments have always favoured the development of custom-made software applications to fill specific necessities as an industrial operations support. Production data collection systems using either databases or spreadsheets are commonly developed on the shop floor to monitor and control real-time and variable execution processes. Software maintenance and data consolidation is obviously complex in such an environment as the number and the structure of these small applications vary over time.

Fortunately, the difficulty of integrating multiple point systems has brought software providers to package multiple execution management components into single and integrated solutions. These systems, commonly referred as Manufacturing Execution System (MES), provide a common user interface and data management system. The MES concept was born from the demand on the manufacturing enterprise to fulfil the requirements of markets from a point of view of reactivity, quality, respect of standards, reduction in cost and deadlines. As such, the functions of MES are primarily turned towards manufacturing activities, a major vector of the added value of all manufacturing firms. Historically well implemented in process industries (pharmaceutical, food and semiconductors industries) where MES systems answer the needs of traceability imposed by the authorities, they are now used in most manufacturing industries.

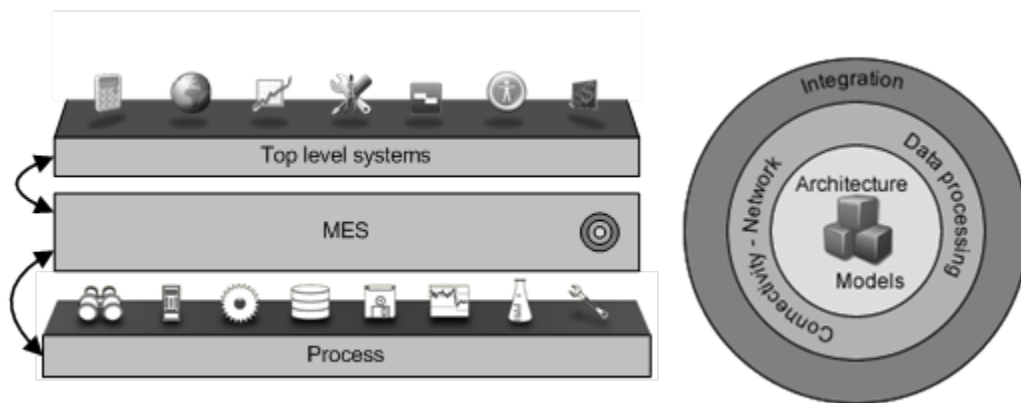


Figure A.1 – environment and its layers

This paper intends to present the actual state of the commercial MES solutions available on the market and to identify major issues related to their use and implementation. We also discuss research areas that need to be explored in order to resolve the increased complexity of such systems. The remainder of the paper is structured as follows. Section 2 first gives a more detailed description of the MES concept followed by a discussion of the technology trends in Section 3. Section 4, 5 and 6 then cover major issues related to MES such as architecture, connectivity and network, and data processing. Section 7 presents integration models proposed by the research

community. These four sections can be viewed as the core, the nerves and the skin of an MES as represented by figure A.1. Finally, we present the current limitations of MES and discuss research areas that need to be explored to fulfil continuing customer needs for faster real-time response and expanded functionality coverage in Section 8; our concluding remarks are in Section 9.

## A.2 Background

Historically, each software editor had their own definition of an MES which was generally based on the capacities of their own tools or on the expectations of their customers. The MESA organization made the first step toward the MES standardization by gathering the major actors of the market and by proposing a formal definition of MES (MESA, 1997c):

MES deliver information that enables the optimization of production activities from order launch to finished goods. Using current and accurate data, an MES guides, initiates, responds to, and reports on plant activities as they occur. The resulting rapid response to changing conditions, coupled with a focus on reducing non value-added activities, drives effective plant operations and processes. The MES improves the return on operational assets as well as on-time delivery, inventory turns, gross margin, and cash-flow performance. An MES provides mission-critical information about production activities across the enterprise and supply chain via bidirectional communications.

To meet the needs of a variety of manufacturing environments, MESA identified 11 principal MES functions (MESA, 1997b). They are as follows (see Figure A.2):

- Operations/Detail Scheduling: sequencing and timing activities for optimized plant performance based on finite capacities of the resources.
- Resource Allocation and Status: guiding what people, machines, tools, and materials should do, and tracking what they are currently doing or have just done.

- Dispatching Production Units: giving the command to send materials or orders to certain parts of the plant to begin a process or step.
- Document Control: managing and distributing information on products, processes, designs, or orders, as well as gathering certification statements of work and conditions.
- Product Tracking and Genealogy: monitoring the progress of units, batches, or lots of output to create a full history of the product.
- Performance Analysis: comparing measured results in the plant with goals and metrics set by the corporation, customers, or regulatory bodies.
- Labour Management: tracking and directing the use of personnel during a shift based on qualifications, work patterns, and business needs.
- Maintenance Management: planning and executing appropriate activities to keep equipment and other capital assets in the plant performing to goal.
- Process Management: directing the flow of work in the plant based on planned and actual production activities.
- Quality Management: recording, tracking, and analyzing product and process characteristics against engineering ideals.
- Data Collection/Acquisition: monitoring, gathering, and organizing data about the processes, materials, and operations from people, machines, or controls.

Among these 11 functionalities, some are directly linked to the process, such as scheduling and quality control, while others, such as resource management and traceability, are best described as cross functions. This difficulty in classifying MES functions has two consequences. First, the concept of MES is often difficult to perceive clearly by the concerned manufacturers. Second, we tend to bring in everything that is not already assigned to other areas, such as ERP. This latter result does not facilitate its structure. In reality, all MES functions are not on the same level.

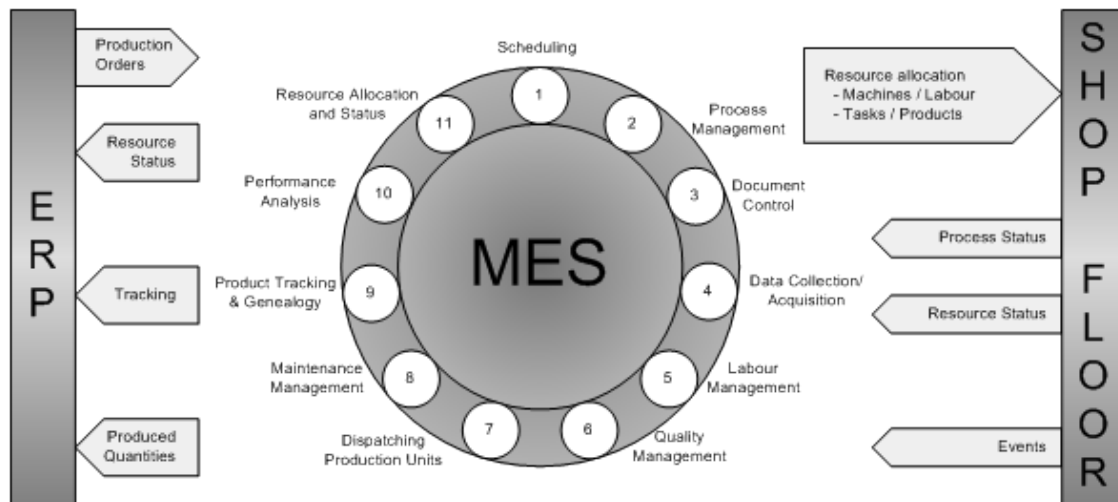


Figure A.2 – MES functionalities

The major contribution of the S95 standard, published by the Instrumentation, Systems & Automation Committee (ISA), is the identification of critical interactions between different parts of an MES system. In the absence of such a system, developed solutions may not be able to follow the changing needs of industry, and may even be unable to effectively respond to the problem. For example, a periodical quality problem may be correctly interpreted if it can be mapped to maintenance operations. Product traceability is inefficient if it is not coupled with the team management on the one hand and process control information on the other.

The standard S95 deals with the formalization of exchange around the manufacturing system to other areas of the company. Designed to be applied to all manufacturing strategies, it does not impose any organizational model or manufacturing system architecture. It suggests a physical model for the company extrapolated from the standard S88 and a definition of functions and information flows based on the Purdue University Reference Model (PRM). The function model of the S95 obviously puts the manufacturing control function in a central position. The MES is then an intermediary between the various departments of the company (Figure A.3). This standard is now widely accepted by market players for the design of information flows between



shop floor level applications and those of a higher level. It also allows the industry to have a consistent terminology.

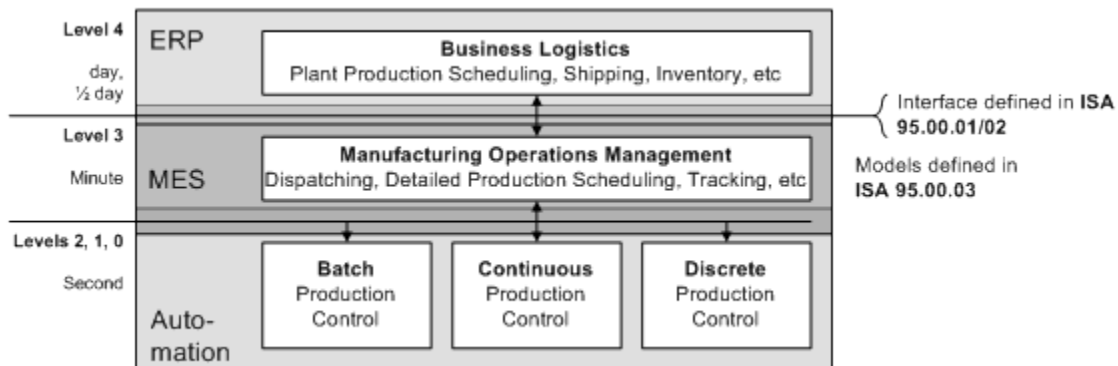


Figure A.3 – MES positioning in the Computer Integrated Manufacturing (CIM) context

All MES functions contribute to achieving on-time performance and total adherence to customer order schedules. According to MESA's survey (MESA, 1997c), MES systems have provided manufacturing enterprises some of the most impressive benefits of any manufacturing software, such as an average 45% reduction in manufacturing cycle time, a significant improvement of the flexibility to respond to customer demands, the realization of certain degrees of agile manufacturing, and customer satisfaction.

### A.3 Technologies

MES solutions are traditionally designed on a modular basis. Thus, if the MES functions set is theoretically invariable, the priority between them and the configuration can vary and be adapted according to the aims of the company. Some functions are sometimes redundant to those of other systems (mainly with the ERP) but are more focused on the production equipment performances. A good example of this modular concept is InSite™, the Camstar MES (Camstar, 2006b), which is composed of 18 integrated modules to collect data and continuously control production

activities. In a general way, most MES modules can be classified in 3 layers (Camstar, 2006a, FactoryTalk, 2006, Proficy, 2006, Simatic IT, 2006, SiView, 2006):

- *Client/server application layer*: This is the layer that is visible to the users. It is also where other systems can be “plugged in,” enabling them to communicate in a distributed messaging environment.
- *Integration framework layer*: The integration framework layer is the heart of the architecture. It is made up of standardized, reusable, object components that provide the application infrastructure through which information flows among applications. This infrastructure is mostly built on standard distributed object tools such as CORBA or COM/DCOM/OLE.
- *Data storage/management layer*: This layer provides essential services for the application and integration framework layers, such as network communications, object management services, and persistent object services. These services and facilities must be built on robust, standard operating system and database technologies to provide a solid and long-term foundation for the whole system.

More specifically, the leading MES companies are focusing on various MES enhancement issues, such as integration, scalability, and graphical configuration (MESA, 1997c). Such improvements are based on the significant advances of network computing (i.e., network protocols, Internet, intranet, and extranet) and software technologies (object, components, client/server architecture, CORBA, ActiveX, Agent, XML, XSLT, SOAP, Web services, universal equipment connectivity, and languages). The improvements and the future business characteristic trends can be summarized as follows:

- Integration: the key to the deployment of an enterprise-wide information system, where an MES bridges the shop floor control and office planning systems.
- Heterogeneous knowledge synthesis: mechanisms to make the collected data more understandable and to make the decision making easier for the user.

- Standardized connectivity: a plug-and-play connectivity for equipments suitable for different computing environments.
- Real-time performance: pertinent information such as the status of process, material, resource, personnel, and service delivered to the right user at the right time.
- Web-based architecture: open and modular system architecture to improve the interoperability for heterogeneous applications and adaptability for the advances of new technologies.
- Scalability and reconfigurability: a MES must take into account the needs of current companies, their size and their dynamic nature.

## **A.4 Architectures and Models**

Scientific literature has mainly proposed models and architectures for developing a general-purpose and efficient MES. In this section, we review some MES open frameworks to facilitate their integration with the current systems and we also discuss the interaction among multiple MES and the data modelling efforts.

Network technology introduces new possibilities such as client-server architecture, parallel, distributed or concurrent working. It generates important challenges like e-business, B2B or B2C, which go together with global competition. Network applications are currently being developed at all levels of manufacturing engineering, issuing new concepts such as agile manufacturing, virtual enterprises, and virtual factories (Jain, 1995). Complexity of production increases as products are getting more and more diversified and suited to customers' requirements and as manufacturing operations and tools are getting more and more sophisticated. Here, the strongly centralized organization is a structural limitation for managing the complexity implied by the combinatorial expansion. Multi-Agent technology applied to manufacturing systems has received particular attention (Van Dyke, 1998, Shen and Norrie, 1999) because of their distributed organization, high modularity and simplicity of implementation. Agents may represent single resources (a work cell, a machine, a tool), workers involved in a process; etc., as well as products,

customers or providers. All together, they perform individual tasks, with interaction among them, in order to fulfil production functions (such as procurement, local planning, task assignment, scheduling, execution control, dispatching, etc.). One of the most fruitful and widespread approaches formalizing the use of agents in manufacturing systems has been developed in the Holonic Manufacturing Systems (Valckenaers *et al.*, 1994), where a “holon” is defined as the association of a software agent with a physical device or a set of physical devices. Although the cooperation of holons does not use either global or centralized procedures, their particular architecture maintains a hierarchical structure based on a functional decomposition. Multi-agent architectures usually use hierarchical and heterarchical structures (Dilts *et al.*, 1991, Leitão and Restivo, 1999). In a hierarchical architecture, there are multiple levels of master-slave agent type relationships, whereas in a heterarchical architecture, agents communicate as “peer-to-peer” without a predefined master-slave relationship. Agents are locally autonomous and cooperate in a negotiation procedure in order to reach their local objective. The advantages expected from heterarchical structures are their reduced complexity, enhanced modularity and scalability, and their intrinsic robustness and fault tolerance. Decisions are made locally only when needed.

The PABADIS project (Plant Automation BAsed on DIstributed Systems) (PABADIS Group, 2001, Diep *et al.*, 2003) aims at developing a MES based on a networked flexible manufacturing system, designed according to a distributed and de-centralized framework, and implemented with autonomous agents. It addresses flexible manufacturing units with small production batches, and will enable future plants to be highly reactive to a turbulent market. Holons are also used in a MES architecture dedicated to mass customized manufacturing (Simao *et al.*, 2006). SEMATECH (SEMATECH Inc., 1998) is another open framework for the integration of MES running in semiconductor industries and focused on the control at the shop level of manufacturing systems. It incorporates basic manufacturing functionalities at the factory engineering level, which include product design, process planning, and material requirement planning (Harhalakis *et al.*, 1994, Harhalakis *et al.*, 1995). It comes with analysis and design tools based on UML diagrams and Petri nets (Lin *et al.*, 2005, Lin and Jeng, 2006). In the same frame of thought, we can talk about NIIP-SMART (Barry *et al.*, 1998) architecture design with a

standards-oriented configurable object model that represents the diverse aspects of MES in order to reduce design costs, or OpenMES (Hori *et al.*, 1999) which is an object-oriented framework with a modular organization to achieve open connectivity not only with other software systems, but also with manufacturing devices.

Most MES research papers report the construction of a single MES and seldom consider its interactions with multiple MES's that represent multiple shop floors. Now, collaborative business has become common and outsourcing functions in enterprises gradually play an important role. Terms such as extended enterprises, contracting organization, and virtual enterprises have been coined for investigating the special cooperative behaviours among companies. However, research usually focuses on the study of value chains and strategic alliances. Little research is done on the execution level. Huang (Huang, 2002) proposes a distributed workflow model based on cooperation types and operation workflow. But there are major limitations because of the different data models used in the distributed MES.

Since the database is the core in an information system, modelling efforts are not only focused on the architectural aspect. The key steps to designing, building and implementing a cost-effective, integrated MES to plant needs are to assess the present state of integration and the availability of infrastructure and then to map the information needs of all the operational layers. Based on this, the database of the MES has to be designed for the plant-specific conditions. Although the third part of the S95 standard attempts to describe the data model of MES, some authors do not await its publication to make proposals. For example, Bing-hai *et al.* (Zhou *et al.*, 2005) adopt the extended entity-relationship model technique to build the data model of MES.

## **A.5 Connectivity and network**

The demand for automation of manufacturing processes is rather high, and most manufacturers make efforts to adopt and/or improve the technique of equipment integration and communications. Yang *et al.* (Yang *et al.*, 1999, Cheng *et al.*, 2000b) have developed a generic equipment manager (GEM) which can either operate alone or be integrated with the MES. In

order to achieve generality and reusability, the governing rules of the GEM are divided into three independent portions: System/Framework Rules, Business Rules, and Equipment Rules. In the first release, these rules were implemented in the MES interface and equipment driver modules separately. Among them, equipment rules in the driver modules are equipment-dependent. Connecting a different machine to the equipment manager requires modifying the associated driver. To solve this problem, the authors have proposed a web-enabled equipment driver that uses mobile object technology. The equipment manager downloads the driver via a web browser, and subsequently, it can communicate with the equipment by either the Java socket technology or the object web technology through the CORBA or DCOM protocols (Cheng *et al.*, 2000a).

Ye (Ye and Qiu, 2003) and Qiu (Qiu, 2003) define generic equipment connectivity as a platform capable for continuous customization. By loading configurable equipment connectivity components and an equipment characteristic descriptor, generic equipment connectivity is then enabled to support proper communication protocols. A schema based on the equipment description is used as an effective vehicle for assimilating different information structures and semantics and to extract pertinent information among overwhelming data from heterogeneous sources.

Networked controlled systems should integrate all new technologies such as wireless networks, embedded systems, nomad components and electronic tags (RFID), to meet new requirements such as mobility, modularity, control and diagnosis decentralization or distribution, autonomy and redundancy, allowing quick and easy maintenance. A major industrial communication challenge of the related multilevel communication architectures is to unify plant networking with Ethernet. The resulting challenge is to guarantee the same deterministic features as those of more specific fieldbuses currently applied in a shop floor. For example, applying fault detection and isolation/fault tolerant control techniques to networked control systems should improve safe control and monitoring of such complex systems as well as their global reliability, dependability and availability by dynamically accommodating network performance, reconfiguring network components and adapting the application to the delivered quality of service (Georges *et al.*, 2006).

## A.6 Data processing

As mentioned previously, data and all that surrounds them are the core of the effectiveness of an MES system. It needs to analyze and put data into context for the user. Like information flows before material does and if the information within a manufacturing enterprise is slow and of poor quality, how can the quality and high-velocity production be ensured? It is clear that having the right information at the right time helps to make the right decisions. Current technology can ensure the availability of information, but it cannot ensure that the information is accurate and meaningful (Jones *et al.*, 2002). If the precision of acquired information depends on the chain of acquisition implemented, the meaning associated to this information only depends on the analysis that we make (Slater, 2000, Rockwell, 2006).

One possibility for gathering data for analysis is to extract data directly from one or more operational databases. Operational or transactional databases are designed to process individual transactions quickly and efficiently. This type of transaction-based interaction is known as on-line transactional processing (OLTP). When transactional data is no longer of value to the operational environment, it can be stored in a data warehouse to be used by a decision support environment. An entity-relationship model is often adopted to create the data models for OLTP system. However, multi-dimensional data models with a star or snowflake (Wrembel and Koncilia, 2006) are adopted as modelling techniques for a decision-based data warehouse. On-line analytical processing (OLAP) is a query-based methodology that supports data analysis in a multidimensional environment. An OLAP engine logically structures multidimensional data in the form of a cube. MS Excel provides an interface that allows us to view OLAP cubes. Thus, OLAP cubes and MES data warehouse can be used to analyze shop floor data. Chen and Wu (Chen and Wu, 2005) design OLAP cubes for analyzing equipment performance, WIP and quality.

On the shop floor, data collected from equipment, devices, and other instrumentations are normally at the necessary level of granularity for fine-tuning manufacturing processes. For instance, they are widely used for statistical process control and run-to-run process adjustments

(Xu and Qiu, 2004). However, if a higher office level information system needs to use this information, they will most likely be of no value to a user without being filtered and put into the context of the relevant business unit domain (Baliga, 1998, Slater, 2000, Qiu *et al.*, 2002). To filter data, standardized data processing technologies, such as document object model (DOM) can be used (Qiu, 2002, Ye and Qiu, 2003). The extraction rules are application/module-dependent but independent of various data sources. Using XML technologies (XML Core Working Group, 2003) makes the integration dynamic and extensible. In addition, combining XSLT language (XSL Working Group, 2001) and XML may permit to dynamically modify the data object configuration (Xu and Qiu, 2004).

## **A.7 Integration Models**

MES have been researched and developed to narrow the gap between the shop floor and office-planning systems (Owen and Parker, 1999, Slater, 2000, Koch, 2001). Obviously system integration to make shop floor data (e.g., inventory availability, equipment utilization, supplier's status, customer orders, commitments, and production schedules) available in a timely way is fundamental to the realization of an integrated enterprise-wide information system, where effective connectivity to the shop floor is not an option but the cornerstone for the well-integrated enterprise-wide information system (MESA, 2000). The benefit of such an integration will be different for each company but the common factors are visibility of processes, ability to make them more responsive and provision of track and trace capabilities. All manufacturing organizations have an integration solution but in most cases they are paper- or people-based (Adshead, 2007). We can split the integration approach into two major categories: the software integration – thanks to application programming interface (API) or agents – and data integration.

Currently, most office information systems are not designed to work at the shop floor level. Howells (Howells, 2000) observed that these systems began as financial applications, moved into human resource, logistics, marketing, and supply chain applications, and often as an afterthought, attempted to link to shop floor manufacturing systems. Lacking a comprehensive design-for-integration consideration in the design stage, office information systems can hardly be connected



to the shop floor. When there is a need, the point-to-point API approach has consequently been adopted by most manufacturing enterprises, giving rise to extremely costly, non-flexible, non-scalable, and hardly sustainable system architecture (Linthicum, 1999, Slater, 2000, Koch, 2001).

After the development of large-scale, non-open and owner solutions, the interfaces' definitions such as the BAPI (Business Programming Application Interfaces) of SAP R/3 became a way of integration in the heterogeneous systems. Because of the need to combine the best modules of various vendors, the concept of reusable preset components was developed and applied to integrate various systems within the same company or between organizations to develop their coordination (Mustafa and Mejabi, 1999). With this kind of interface, it is possible to be integrated in the internal processes of the software. For example, the maintenance models – which are the subject of much research – are often integrated into the ERP. Ip *et al.* (Ip *et al.*, 2000) use the integrated definition method (IDEF) model to systematically integrate maintenance into MRPII. Whereas Nikolopoulos *et al.* (Nikolopoulos *et al.*, 2003) have preferred the object approach, Rizzi and Zamboni (Rizzi and Zamboni, 1999) developed a module integrated into an ERP to optimize the perishable goods storage.

As in the PABADIS project, mobile agents can also be used to model MES architectures. But the capacities of agents to negotiate in an autonomous way to succeed in achieving the goal which is assigned to them can also be used as integration vector. They were already implemented to manage the capacity of a shop floor from an ERP (Mertens *et al.*, 1994) or to incorporate shop-floor controls formally into plant-wide information-control systems for enabling “on the fly” rescheduling of product routes as well as manufacturing process reconfigurability (Tang and Qiu, 2004). The development of this method is made easier by the JAVA programming language capabilities, which natively implement the mobile agents and are able to transform the complex architecture of an agent into simple text file and vice-versa (Wong *et al.*, 1999).

Howells also pointed out in (Howells, 2000), that the reality of manufacturing businesses was based on the facts that:

- Regardless of the products manufactured, a financial package can be implemented in any industry.
- Regardless of the products manufactured, most logistics requirements are similar.
- It is in the manufacturing plant that the unique differentiators and business data exist.

In other words, different manufacturing equipments (i.e., processes) are used in making different products, such as automobiles, food, chemicals, etc. It is the variety of types of manufacturing equipment that complicates the shop floor connectivity. At the present time, data are transported from various sources in various formats using various communication interfaces. Efficient data integration will be best suitable for the integration architecture. For example, Wong (Wong, 2006) proposes a data model to combine an ERP and a PLM. With S95 standard, the goal of shop floor to “top floor” integration is easier to achieve. Integration has been done for many years with CVS files and spreadsheet macros resulting in non-flexible integration. Now, things are relatively standard to connect the shop floor with MES and ERP systems. For example, OPC (OLE for Process Control) can gather data from control systems and B2MML (the XML implementation of S95 standard) and get it up to ERP application (Adshead, 2007). XML standard is the ideal approach to data harmony: the integration becomes dynamic, extensible and can be dynamically modified (Salam *et al.*, 1999).

Lobecke and Slawinski (Lobecke and Slawinski, 2004) also discuss the integration of an MES and an ERP. It turns out that a consistent data model is essential for the horizontal and vertical integration of the systems.

## **A.8 Limitations of current MES and challenges**

Limitations referred to here mainly concern the practical environment where the MES operates. We discuss the three layers represented in Figure A.1: the *architectural* aspect, the *connectivity*

impact on the real-time data processing and the *integration* difficulties of such systems. We also evoke some tracks proposed in the literature to reduce the impact of those limitations and some challenges for MES evolution.

### **A.8.1 Architecture**

Today, the major architectural issue is the lack of tools and platforms to test and validate new architectural developments on realistic problems, in terms of both the size of the manufacturing system and the thoroughness of the evaluation techniques. Recently, research on applying multi-agent systems in manufacturing has produced many results (Muhl *et al.*, 2003). However, various obstacles remain for deployment in industry. Marik and Lazansky (Marik and Lazansky, 2007) emphasize that only very few real-life industrial experiments are in effect, despite laboratory experiments on the promising MAS and HMS approaches. Therefore, an environment is required in which the research community can provide and retrieve test cases of realistic size and complexity; in other words, research developments must be tested in real-world emulated factories in place of the toy test cases and token evaluation campaigns which remain the norm.

### **A.8.2 Real-time processing**

Theoretically, an MES must be able to process the data of the shop floor in real time (minutes/hours) to be able to control the manufacturing activities (MESA, 1997c, MESA, 1997b). To be reactive, it must not only obtain these data, but also analyze them in real time. When controls are made in seconds or fractions of seconds in shop floor process execution systems, the MES may take minutes or even hours to respond. Compared with the chaos of old shop floors, where no current and accurate shop floor information was available and ERP was implemented as open-loop systems with response times in days, the current MES solutions are relatively “real time.”

In general, the scalability of a current MES is very much limited in terms of the capability to keep its real-time resolutions when the governed manufacturing system grows. It is evidence that the amount of information collected from control systems increases tremendously with the degree of

increased automation on the shop floor. That is to say, MES are required for processing more information. Manufacturing systems grow because of the need for more complex processes to meet the needs of increased product functionality. In other words, MES are required to be either physically or logically connected to more equipment and are dealing with tremendously more data. It is a vicious circle: the more an MES wants to be closest to the real-time notion, the more it must be connected; the more it wants to be connected, the more it wants to process data; finally, the more it wants to process data, the less it can be closest to the notion of real-time processing.

In addition to being able to improve the productivity of a global enterprise manufacturing system, the deployed MES in such a circumstance has to be capable of obtaining all of the manufacturing process and product data from all plants geographically dispersed around the world in a timely way. These data should also be analyzed in real time to make enterprise-wide optimal decisions such as validating the equipment setup and production schedule, monitoring the process status and indicating the trend of product quality, reacting more quickly to dynamic customer demands when parts of a typical order could be simultaneously made in different plants.

### **A.8.3 Integration**

First of all, establishing a hierarchy of production tasks, which has always been the unique organizational scheme, seems to be supplanted by other forms of management. In conventional systems like MRPII, MES executes and controls production orders which emanate from ERP. This top-down structure does not easily integrate different production forms, such as Just In Time (JIT) production, “pull” production, or inverse manufacturing which favours a bottom-up production demand (Massotte, 1997), and necessitates a change in the processing of orders.

Another difficulty comes from the software heterogeneity of the manufacturing environment. Indeed, it is not rare within a same MES solution to have to implement modules of various providers, each one with its data model, its communication mechanisms and even its own data base. To have them work as a whole, proprietary mechanisms are commonly applied for system integration. Furthermore, because MES applications are often mission-critical while lacking

proper adaptability to new technologies, they may not be replaced or updated as the technology advances.

The adoption of owner formats and API by the providers stays, and shall probably stay, the major reason limiting the integration of these tools. It is why the implementation, the integration and the maintenance of MES are so expensive. The large companies miss time and the small ones miss resources to conclude these projects (Koch, 2001). Although the MES development is a prosperous industry, the high costs of development and the owner formats slow down the adoption of these tools in many companies. By nature and definition MES are ready to be integrated. The problem is that each information system (ERP, SCM, etc.) has owner interfaces, and in particular the manufacturing equipment is innumerable and quite often unique. Academic research on integration problems show it and all are unanimous in stressing the importance of the integration of MES with other information systems. Since 1996, Westerlund (Westerlund, 1996) has insisted on the importance of the integration capacities in the choice of ERP and MES systems because it is one of the keys to achieving financial goals. In 2001, Rondeau and Litteral (Rondeau and Litteral, 2001) reiterated this call. Liu *et al.* (Liu *et al.*, 2002) describe the difficulties in integrating an Advanced Planning System (APS), an ERP and an MES to check the capacity constraints of the production plans.

The Service-Oriented Architecture (SOA) technology can solve those integration issues. SOA is a design for linking computational resources (principally, applications and data) on demand to achieve the desired results for service consumers (who can be end users or other services). This style of architecture promotes reuse at the macro (service) level rather than the micro levels (e.g., objects). It can also simplify interconnection to and usage of existing IT (legacy) assets. SOA is an architectural style which goal is to achieve loose coupling among interacting software agents. A service is a unit of work done by a service provider to achieve desired end results for a service consumer. Both provider and consumer are roles played by software agents on behalf of their owners. The only disadvantage of this solution it is that it would require the complete rewriting of all the current applications (Erl, 2005). But nowadays, many researches and major

contributions have been found on service-driven manufacturing platforms (Chazalet and Lalanda, 2007, Jiang *et al.*, 2007).

Manufacturing execution is a complex task because of the nonlinear nature of the production system, the production processes, and the environmental uncertainties. Schedules and plans, originating from higher levels in a manufacturing organization, can become ineffectual within minutes on a shop floor. Moreover, the variety of existing manufacturing system types and performance issues, as well as the different kinds of equipment and processes, is very wide. To cope with this heterogeneity challenge, future MES designs must apply the most fundamental and recent insights in self-organizing systems, a topic that is being intensely investigated by the multi-agent systems community today (Di Marzo *et al.*, 2004). To design such self-organizing systems, it is also essential to apply insights from fundamental research (Waldrop, 1993, Valckenaers *et al.*, 2003) and to define the related modelling framework to obtain the required system features. Important expected progress in the domain is the emergence of MES's that are able to forecast the emerging state of the manufacturing system while preserving the level of decoupling that has made older multi-agent manufacturing execution systems robust and configurable (Valckenaers *et al.*, 2004).

#### **A.8.4 Research challenges**

For modern companies, processes are the privileged form of action, they are the added value. It is also why they have invested in the integration of their processes into ERP. But until the emergence of new requirements in reactivity, in quality, and in adaptability, the manufacturing processes were forsaken. The objective behind the MES concept is the optimization of the manufacturing processes and resources. The first step is, naturally, the performance measurement of the current system. MES is naturally strongly connected to the manufacturing processes. It is the best tool to measure in real time performance indicators such as the use of materials, the productivity according to batches, and the machine breakdown. Then, it is possible to create piloting dashboard by evaluating indicators such as the synthetic yield, the mean time between failures, or the mean time to repair. Consequently, the actions to be set up appear obvious. Here

we find the steps of the continuous improvement diagram: Measure, Analyze and Improve. Those steps largely use data-processing technologies, but also require human expertise. It is a first step in the decision-making support: the MES alarms, presents and formats the data wished for by the user. But the decision support tools fall not only into information processing but also into operational research and analytical methods. The contribution of those systems can result in intervention to one of six possible levels in the decision-making process (Decide loop of Figure A.4): (1) elementary data storage, (2) the data return, (3) the relevant data selection, (4) the decision model design, (5) the decision evaluation, (6) the selection and choice of a solution.

Lobecke and Slawinski (Lobecke and Slawinski, 2004) present the integration of an MES and an ERP. It turns out that a consistent data model is essential for the horizontal and vertical integration of the systems. When the businesses processes are well integrated in the data model, optimization potential can be exploited. The MES functionality implements a set of optimization methods that rely on a fixed, rigid plant model. Actually, most processes are not or cannot be explicitly modelled, which leads to the occurrence of unexpected events (Homem-de-Mello *et al.*, 1999). The MES integration with the other information systems will make it possible for organizations to quickly respond to disturbing events that arise on the shop floor.

MES applications do not address all of the manufacturing execution processes required to replenish the supply chain while dynamically responding to unpredicted change. But the MES integration with other information systems will make it possible to adopt adaptive manufacturing strategies by quickly answering to disturbing events appearing in the shop floor. Figure 4 presents the adaptive manufacturing strategy loop combined with 6 levels of the decision-making process where grey activities can be supported by MES. While MES is seen as the cornerstone of such approach, it still needs to be combined to SCM, PLM and other information system functions which reinforce the need for adopting a SOA or multi-agent design approach. Also, adaptive manufacturing strategies require defining the process management layer between ERP and process control and translating demand-driven supply chain requirements into a set of capabilities, systems, and workflow integration investments (Martin, 2004). Therefore, adaptive manufacturing organizations need an execution platform that connects manufacturing processes

with enterprise and supply chain processes, enables closed-loop mechanisms, and provides decision support to production personnel so they can deliver on their performance goals.

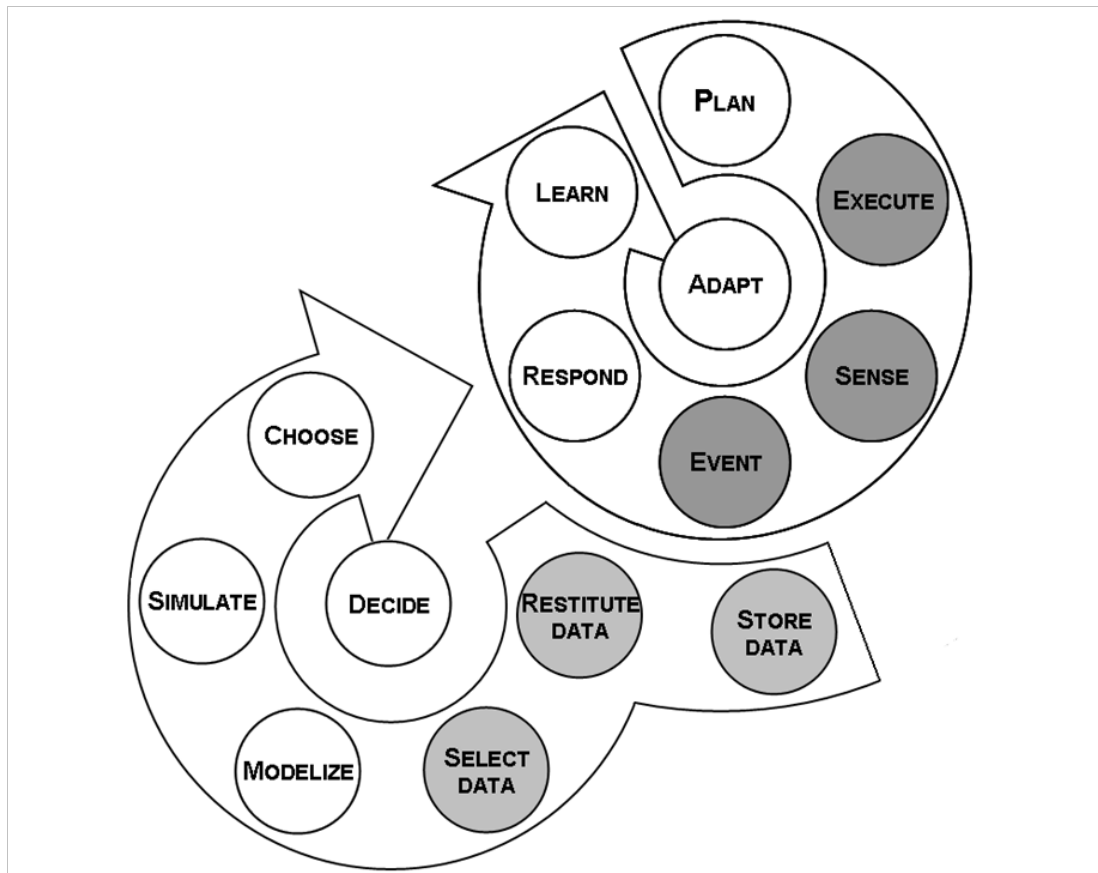


Figure A.4 – Adaptive Intelligent Manufacturing System

In grey: covered by current MES.

In addition, unexplored research issues remain to develop the decision-making support capacities:

- one difficulty is to develop explicative models in order to react in real-time to hazards like defects or failures, and to integrate stochastic treatments;



- the cooperation between high level planners and schedulers and the MES is virtually unexplored;
- scaling the MES technology to multi-site manufacturing coordination and control is only in the initial stages of research.

### **A.8.5 Industrial challenges and opportunities**

As recent surveys (MESA, 1997a, Fraser, 2004) revealed, firms measuring and linking operations and financial performance indicators perform best. Also, surveyed businesses using automated data collection on the shop floor achieved better improvements in quality, throughput, customer service, compliance, asset utilisation and inventory. All these arguments should motivate manufacturers to adopt MES solutions. However, justifying the implementation of MES to improve manufacturing and financial performance is a difficult task for many executives. As MES implementation can bring intangible benefits and involves significant costs and risks, a comprehensive and systematic assessment is necessary before making such a critical investment decision.

Traditionally, system investment projects are evaluated based on financial criteria, such as return on investment, net present value or internal rate of return. This approach may underestimate the value of MES as it considers only tangible and monetary effects and ignore intangible ones such as flexibility and agility. Fortunately, Liang and Li (Liang and Li, 2007) propose a decision methodology based on *Analytic Network Process* to render it facile for a manufacturer to make the decision of whether or not to implement an MES and which solution should be chosen. However, other questions still need to be answered: Which processes should be implemented first? Which data should be collected? Differences in priorities, manufacturing processes, and enterprise environment will certainly lead to different decisions from one organization to another. In all cases, manufacturers should favour processes and data which will generate meaningful performance indicators. Indeed, MES provides visibility to accurate, high-velocity information about production performance at a fraction of the cost and time of an ERP initiative.

An MES implementation project can also become an opportunity to upgrade processes, to recognize and then seize new opportunities both internally and in the marketplace. To achieve this, shop floor staff need more than data analysis. They need decision tools to dynamically respond to unpredicted change. MES is the appropriate application not only to gain access to critical data in real-time but also to propagate the decision made. To do so, MES solutions need to combine different solving tools with multiple data sources to elaborate solutions in reasonable time compatible with the planning horizon. The recent advances in computing power and memory have opened up the possibility of on-line simulation-based optimization. This approach offers one of the most interesting opportunities in simulation and the potential benefits in this field are significant and are just beginning to be exploited. A typical area of application for online simulation, also called real-time simulation, is proactive decision support for scheduling problems in manufacturing systems (Siemiatkowski and Przybylski, 2006). For example, deployment of a decision support system in a plant of Nippon Kokan Co. Ltd. allows to divide by two the metal waiting times, which resulted in a reduction of one million dollars a year in production costs (Numao, 1994). Saenz de Ugarte *et al.* (2006) proposed a real-time execution architecture adding a decision module to the ERP-MES tandem. This architecture is based on efforts already deployed to integrate MES with ERP, and provides a demonstrative example based on a real manufacturing scenario in the aluminium industry.

## A.9 Conclusions

This paper presented a literature review of Manufacturing Execution Systems (MES) which had been developed some years ago to fill a gap between shop floor information and control systems and other enterprise information systems. It presented different approaches proposed by the scientific community in order to facilitate its implementation and integration in complex production environment. We also discussed the limitations of the current MES and the challenges for the future of this system.

Despite all the advances in this area, we note that this industry and the research community have mostly addressed the manufacturing execution problem from a systemic point of view. However,

if we want manufacturers to move from a “react to forecast changes” mindset to an “adaptive” mindset, researchers must support them by developing new models and approaches which will proactively detect production exceptions and provide production personnel with on-line decision support with the objective of bringing the factory and its supply chain to react to the moving targets imposed by its customers.

Indeed, potential research areas are numerous, varying from the development of new re-scheduling algorithms to the development of adaptive performance indicators. On-line decision making tools are important and should integrate evaluation methods. Simulation remains a powerful tool for evaluating different scenarios and calculating performance indicators for every considered strategy. The combination of on-line simulation and optimization algorithms (mainly heuristics) is now quite established in the academic world but a lot still remains to be achieved in industry. Data mining tools would also help in explaining information gathered at different levels. Aggregating efficiently data to have consistent and significant global performance indicators and their interaction with lower levels indicators continues to be an interesting research topic.

## A.10 References

- Adshead, A., 2007. From shopfloor to top floor manufacturing execution systems. *Manufacturing Computer Solutions*, 13(2), 44-5.
- Baliga, J., 1998. MES and CIM: at the center of productivity. *Semiconductor International*, 21(8), 104-106, 108, 110, 112.
- Barry, J., Aparicio, M., Durniak, T., Herman, P., Karuturi, J., Woods, C., Gilman, C., Lam, H. & Ramnath, R., 1998. NIIP-SMART: an investigation of distributed object approaches to support MES development and deployment in a virtual enterprise. *Proceedings of Second International Enterprise Distributed Object Computing Workshop, 1998. EDOC '98*, 366-377.
- Camstar, 2006a. *MES automation - Amkor technology - IC assembly* [online]. Camstar. Available from: <http://www.camstar.com/documentation.aspx> [Accessed 2 February 2007].

Camstar, 2006b. *Products : MESA & InSite* [online]. Camstar. Available from: <http://www.camstar.com/documentation.aspx> [Accessed 2 February 2007].

Chazalet, A. & Lalanda, P., 2007. Deployment of services-oriented applications integrating physical and IT systems. *21st International Conference on Advanced Information Networking and Applications, AINA 2007*, Niagara Falls, ON, Canada: Institute of Electrical and Electronics Engineers Inc. Piscataway NJ 08855-1331 United States, 38-45.

Chen, K.-Y. & Wu, T.-C., 2005. Data warehouse design for manufacturing execution systems. *2005 IEEE International Conference on Mechatronics, ICM '05*, Taipei, Taiwan: Institute of Electrical and Electronics Engineers Computer Society Piscataway NJ 08855-1331 United States, 751-756.

Cheng, F.-T., Lin, M.-T. & Lee, R.-S., 2000a. Developing a web-enabled equipment driver for semiconductor equipment communications. *Proceedings of the ICRA 2000: IEEE International Conference on Robotics and Automation*, San Francisco, CA: Institute of Electrical and Electronics Engineers Inc. Piscataway NJ USA, 2203-2210.

Cheng, F.-T., Yang, H.-C., Kuo, T.-L., Feng, C. & Jeng, M., 2000b. Modeling and analysis of equipment managers in manufacturing execution systems for semiconductor packaging. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 30(5), 772-782.

Di Marzo, S.G., Karageorgos, A., Rana, O.F. & Zambonelli, F., 2004. Engineering self-organising systems: nature-inspired approaches to software engineering. *Lecture Notes in Artificial Intelligence*, Verlag Heidelberg D-69121 Germany: Springer, 299.

Diep, D., Massotte, P. & Meimouni, A., 2003. A distributed manufacturing execution system implemented with agents: the PABADIS model. *Proceedings. IEEE International Conference on Industrial Informatics*, Banff, Alta., Canada: IEEE, 301-6.

Dilts, D.M., Boyd, N.P. & Whorms, H.H., 1991. The evolution of control architectures for automated manufacturing systems. *Journal of Manufacturing Systems*, 10(1), 79-93.

Erl, T., 2005. *Service-Oriented Architecture: Concepts, Technology, and Design* Upper Saddle River: Prentice Hall PTR.

Factorytalk, 2006. *Integrate Your Manufacturing and Enterprise Information* [online]. Rockwell Automation. Available from: <http://www.rockwellautomation.com/rockwell-software/factorytalk/> [Accessed 2 February 2007].

Fraser, J., 2004. *The MES Performance Advantage: Best of the Best Plants Use MES* [online]. Rockwell Automation, Inc. Available from: [http://domino.automation.rockwell.com/-applications/gs/region/gtswebst.nsf/files/MES\\_Perf\\_Adv.pdf/\\$file/MES\\_Perf\\_Adv.pdf](http://domino.automation.rockwell.com/-applications/gs/region/gtswebst.nsf/files/MES_Perf_Adv.pdf/$file/MES_Perf_Adv.pdf) [Accessed 12 December 2007].

Georges, J.-P., Krommenacker, N., Divoux, T. & Rondeau, E., 2006. A design process of switched Ethernet architectures according to real-time application constraints. *Engineering Applications of Artificial Intelligence*, 19(3), 335-344.

Harhalakis, G., Lin, C.-P., Mark, L. & Muro-Medrano, P., 1994. Implementation of rule-based information systems for integrated manufacturing. *IEEE Transactions on Knowledge and Data Engineering*, 6(6), 892-908.

Harhalakis, G., Lin, C.P., Mark, L. & Muro-Medrano, P.R., 1995. Structured representation of rule-based specifications in CIM using updated petri nets. *IEEE Transactions on Systems, Man and Cybernetics*, 25(1), 130-144.

Holst, L., 2001. Integrating Discrete-Event Simulation into the Manufacturing System Development Process. Lund University, Department of Mechanical Engineering, Division of Robotics.

Homem-De-Mello, T., Shapiro, A. & Spearman, M.L., 1999. Finding optimal material release times using simulation-based optimization. *Management Science*, 45(1), 86-101.

- Hori, M., Kawamura, T. & Okano, A., 1999. OpenMES: scalable manufacturing execution framework based on distributed object computing. *1999 IEEE International Conference on Systems, Man, and Cybernetics 'Human Communication and Cybernetics'*, Tokyo, Jpn: Institute of Electrical and Electronics Engineers Inc. USA, 398-403.
- Howells, R., 2000. ERP needs shop-floor data. *Manufacturing Engineering*, 125(4), 54-62.
- Huang, C.-Y., 2002. Distributed manufacturing execution systems: A workflow perspective. *Journal of Intelligent Manufacturing*, 13(6), 485-497.
- Ip, W.H., Kwong, C.K. & Fung, R., 2000. Design of maintenance system in MRPII. *Journal of Quality in Maintenance Engineering*, 6(3), 177-191.
- Jain, S., 1995. Virtual factory framework: a key enabler for agile manufacturing. *Proceedings of the 1995 INRIA/IEEE Symposium on Emerging Technologies and Factory Automation. Part 1 (of 3)*, Paris, Fr: IEEE Piscataway NJ USA, 247-258.
- Jiang, P.Y., Zhou, G., Zhao, G., Zhang, Y.F. & Sun, H.B., 2007. e<sup>2</sup>-MES: an e-service-driven networked manufacturing platform for extended enterprises. *International Journal of Computer Integrated Manufacturing*, 20(2-3), 127-42.
- Jones, A.T., Reeker, L.H. & Deshmukh, A.V., 2002. *On Information and Performance of Complex Manufacturing Systems* [online]. [http://www.ifm.eng.cam.ac.uk/mcn/pdf-\\_files/part6\\_1.pdf](http://www.ifm.eng.cam.ac.uk/mcn/pdf-_files/part6_1.pdf) [Accessed 8 october 2007].
- Koch, C., 2001. Why your integration efforts end up looking like this... *CIO Magazine*, 15, 98-108.
- Leitão, P. & Restivo, F., 1999. A layered approach to distributed manufacturing. *Proceedings of ASI'99 International Conference*, Leuven, Belgium, 21-23.
- Liang, C. & Li, Q., 2007. Manufacturing Execution Systems (MES) assessment and investment decision study. *2006 IEEE International Conference on Systems, Man and Cybernetics*, Taipei, Taiwan: Institute of Electrical and Electronics Engineers Inc. New York NY 10016-5997 United States, 5285-5290.

- Lin, C.-P., Jeng, L.-D., Lin, Y.-P. & Jeng, M., 2005. Management and control of information flow in CIM systems using UML and Petri nets. *International Journal of Computer Integrated Manufacturing*, 18(2-3), 107-121.
- Lin, C.-P. & Jeng, M., 2006. An expanded SEMATECH CIM framework for heterogeneous applications integration. *IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans*, 36(1), 76-90.
- Linthicum, D.S., 1999. *Enterprise Application Integration* MA: Addison-Wesley.
- Liu, W., Chua, T.J., Lam, J., Wang, F.Y., Cai, T.X. & Yin, X.F., 2002. APS, ERP and MES systems integration for Semiconductor Backend Assembly. *Proceedings of the 7th International Conference on Control, Automation, Robotics and Vision, ICARCV 2002*, Marine Mandarin, Singapore: Nanyang Technological University Singapore, Singapore, 1403-1408.
- Lobecke, R. & Slawinski, T., 2004. Integrated manufacturing execution systems - history and current state: from system integration to overall optimisation of business processes. *Automatisierungstechnische Praxis*, 46(1), 21-5.
- Marik, V. & Lazansky, J., 2007. Industrial applications of agent technologies. *Control Engineering Practice*, 15(11), 1364-1380.
- Martin, R., 2004. *Produce to Demand Is a Manufacturing Strategy, Not an MES Project* [online]. AMR Research. Available from: <http://www.amrresearch.com/Content/-View.asp?pmillid=17201> [Accessed 24 october 2007].
- Massotte, P., 1997. Analysis and approaches for the management of complex production systems. In A. Artiba & S.E. Elmaghraby, eds. *The Planning and Scheduling of Production Systems*. London: Chapman & Hall, 89-131.
- Mertens, P., Falk, J. & Spieck, S., 1994. Comparisons of agent approaches with centralized alternatives based on logistical scenarios. *Information Systems*, 19(8), 699-709.

Mesa, 1997a. *White Paper #01: The Benefits of MES: A Report from the Field* [online]. MESA International. Available from: <http://www.mesa.org/knowledge-base/details.php?id=48> [Accessed 12 december 2007].

Mesa, 1997b. *White Paper #02: MES Functionalities and MRP to MES Data Flow Possibilities* [online]. MESA International. Available from: <http://www.mesa.org/knowledge-base/details.php?id=49> [Accessed 5 october 2007].

Mesa, 1997c. *White Paper #06: MES Explained: A High Level Vision for Executives* [online]. MESA International. Available from: <http://www.mesa.org/knowledge-base/details.php?id=53> [Accessed 5 october 2007].

Mesa, 2000. *White Paper #03: The Controls Layer: Controls Definition and MES to Controls Data Flow Possibilities* [online]. MESA International. Available from: <http://www.mesa.org/knowledge-base/details.php?id=50> [Accessed 5 october 2007].

Muhl, E., Charpentier, P. & Chaxel, F., 2003. Optimization of physical flows in an automotive manufacturing plant: Some experiments and issues. *Engineering Applications of Artificial Intelligence*, 16(4), 293-305.

Mustafa, Y. & Mejabi, O., 1999. An approach for developing flexible MRP systems. *Information Systems Management*, 16(2), 58-63.

Nikolopoulos, K., Metaxiotis, K., Lekatis, N. & Assimakopoulos, V., 2003. Integrating industrial maintenance strategy into ERP. *Industrial Management and Data Systems*, 103(3-4), 184-191.

Numao, M., 1994. Development of a cooperative scheduling system for the steel making process. In M. Zweben & M.S. Fox, eds. *Intelligent Scheduling*. San Fransisco: Morgan Kaufmann Publishers.

Owen, T. & Parker, K., 1999. One app's ceiling is another app's floor. *Manufacturing Systems*, 17(10), 6 pp.



Pabadis Group, 2001. *Revolutionizing Plant Automation : the PABADIS Approach* [online]. Deliverable 6.3. Available from: [http://www.pabadis.org/downloads/pabadis\\_del\\_6\\_3.pdf](http://www.pabadis.org/downloads/pabadis_del_6_3.pdf) [Accessed 12 November 2006].

Proficy, 2006. *Proficy, Plant Applications* [online]. GE Fanuc. Available from: [http://www.gefanuc.com/Downloads/en/proficyplantapp\\_cutsheet\\_gfa594.pdf](http://www.gefanuc.com/Downloads/en/proficyplantapp_cutsheet_gfa594.pdf) [Accessed 8 october 2007].

Qiu, R., Burda, R. & Chylak, R., 2002. Distributed WIP control in advanced semiconductor manufacturing. *Proceedings of the 13th Annual IEEE/SEMI Advanced Semiconductor Manufacturing Conference*, Boston, MA, United States: IEEE Inc., 49-54.

Qiu, R.G., 2002. A data fusion framework for an integrated plant-wide information system. *Proceedings of 5th International Conference on Information Fusion*, Annapolis, MD: Int. Soc. Inf. Fusion, 101-7 vol.1.

Qiu, R.G., 2003. E-manufacturing: The keystone of a plant-wide real time information system. *Journal of the Chinese Institute of Industrial Engineers*, 20(3), 266-274.

Rizzi, A. & Zamboni, R., 1999. Efficiency improvement in manual warehouses through ERP systems implementation and redesign of the logistics processes. *Logistics Information Management*, 12(5), 367-77.

Rockwell, 2006. *Making sense of e-manufacturing: A roadmap for manufacturers* [online]. Rockwell Automation. Available from: <http://www.automation.rockwell.com> [Accessed 2 February 2007].

Rondeau, P.J. & Litteral, L.A., 2001. Evolution of manufacturing planning and control systems: From reorder point to enterprise resource planning. *Production and Inventory Management Journal*, 42(2), 1-7.

- Saenz De Ugarte, B., Artiba, A., Jbaida, K. & Pellerin, R., 2006. Adaptive Manufacturing: A Real-Time Simulation-Based Control System. *Proceedings of the International Conference on Software Engineering Advances (ICSEA 2006)*, Papeete, Tahiti, French Polynesia: IEEE Computer Society, 71-77.
- Salam, A.F., Rao, H.R. & Bhattacharjee, S., 1999. Internet-based technologies: value creation for the customer and the value chain across industries. *Proceedings of AIS AMCIS 99: 1999 Americas Conference on Information Systems*, Milwaukee, WI: Assoc. Inf. Syst, 538-40.
- Sematech Inc., 1998. *Computer Integrated Manufacturing Framework Specification Version 2.0* Austin, TX.
- Shen, W. & Norrie, D.H., 1999. Agent-based systems for intelligent manufacturing: a state-of-the-art survey. *Knowledge and Information Systems*, 1(2), 129-56.
- Siemiatkowski, M. & Przybylski, W., 2006. Simulation studies of process flow with in-line part inspection in machining cells. *Journal of Materials Processing Technology*, 171(1), 27-34.
- Simao, J.M., Stadzisz, P.C. & Morel, G., 2006. Manufacturing execution systems for customized production. *Journal of Materials Processing Technology*, 179(1-3), 268-275.
- Simatic It, 2006. *SIMATIC IT Framework - Production Modeling and Component Coordination* [online]. Siemens. Available from: <http://www.sea.siemens.com/-mes/product/msitfram.html> [Accessed 8 october 2007].
- Siview, 2006. *IBM SiView Standard* [online]. IBM. Available from: <http://www-06.ibm.com/jp/iisc/english/po/siview/about.html> [Accessed 8 october 2007].
- Slater, D., 2000. *Talk to Your Plants* [online]. CIO. Available from: [http://www.itworld.com/App/670/CIO031500\\_plants\\_content/](http://www.itworld.com/App/670/CIO031500_plants_content/) [Accessed 8 october 2007].
- Tang, Y. & Qiu, R.G., 2004. Integrated design approach for virtual production line-based reconfigurable manufacturing systems. *International Journal of Production Research*, 42(18), 3803-3822.

- Valckenaers, P., Saint Germain, B., Verstraete, P., Hadeli, Zamfirescu, C. & Van Brussel, H., 2004. Ant colony engineering in coordination and control: how to engineer a short term forecasting system. *Proceedings of the 5<sup>th</sup> International Workshop on Emergent Synthesis, IWES'04*, Budapest, Hungary, 125-132.
- Valckenaers, P., Van Brussel, H., Bonneville, F., Bongaerts, L. & Wyns, J., 1994. IMS test case 5. Holonic manufacturing systems. *Proceedings of the IFAC Workshop on Intelligent Manufacturing Systems (IMS'94)*, Vienna, Aust: Pergamon Press Ltd, 31-36.
- Valckenaers, P., Van Brussel, H., Hadeli, Bochmann, O., Germain, B.S. & Zamfirescu, C., 2003. On the design of emergent systems: An investigation of integration and interoperability issues. *Engineering Applications of Artificial Intelligence*, 16(4), 377-393.
- Van Dyke, H., 1998. What can agents do in industry, and why? An overview of industrially-oriented R&D at CEC. *Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science)*, 1435(1), 1-18.
- Waldrop, M.M., 1993. *Complexity: the emerging science at the edge of order and chaos*. New-York (NY): Touchstone.
- Westerlund, T., 1996. ERP and MES integration: Reducing cycle time. *Proceedings of the 1996 Industrial Computing Conference, ICS/96*, Chicago, IL: Instrument Society of America Research Triangle Park NC USA, 91-96.
- Wong, D., Paciorek, N. & Moore, D., 1999. Java-based mobile agents. *Communications of the ACM*, 42(3), 92-102.
- Wong, K., 2006. We gather today to join ERP and PLM: Marrying enterprise data to product data. *Cadalyst*, 23(9), 42-44.
- Wrembel, R. & Koncilia, C., 2006. *Data Warehouses And Olap: Concepts, Architectures And Solutions* Hershey, PA: IRM Press.
- Xml Core Working Group, 2003. *Extensible Markup Language (XML)* [online]. W3C. Available from: <http://www.w3.org/XML/> [Accessed 8 october 2007].

Xsl Working Group, 2001. *The Extensible Stylesheet Language Family (XSL)* [online]. W3C. Available from: <http://www.w3.org/Style/XSL/> [Accessed 8 october 2007].

Xu, Q. & Qiu, R., 2004. Heterogeneous knowledge syntheses for manufacturing information systems. *Jisuanji Jicheng Zhizao Xitong/Computer Integrated Manufacturing Systems, CIMS*, 10(7), 732-736.

Yang, H.-C., Cheng, F.-T. & Huang, D., 1999. Development of a generic equipment manager for semiconductor manufacturing. *Proceedings of the 1999 7th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA'99)*, Barcelona, Spain: IEEE Inc. Piscataway NJ USA, 727-732.

Ye, S.X. & Qiu, R.G., 2003. An architecture of configurable equipment connectivity in a future manufacturing information system. *Proceedings 2003 IEEE International Symposium on Computational Intelligence in Robotics and Automation*, Kobe, Japan: IEEE, 1144-9.

Zhou, B.-H., Wang, S.-J. & Xi, L.-F., 2005. Data model design for manufacturing execution system. *Journal of Manufacturing Technology Management*, 16(8), 909-935.

## **ANNEXE B: ENGINEERING CHANGE ORDER PROCESSING IN ERP SYSTEMS: AN INTEGRATED REACTIVE MODEL**

Référence : Saenz de Ugarte, B., Hajji, A., Pellerin, R., & Artiba, A. (2009). Engineering Change Order Processing in ERP Systems: An Integrated Reactive Model. *European Journal of Industrial Engineering*. (Accepté)

**Abstract** – Manufacturing Execution System (MES) provides a common and single system to support most of the manufacturing execution processes from the production order release to the delivery of finished goods. However, MES applications do not address all of the manufacturing execution processes required to replenish the supply chain while dynamically responding to unpredicted change and fall short of adding decision-making tools capabilities. To fill this gap, we propose a generic manufacturing execution platform where real-time optimization models complement the traditional ERP/MES model. The main objective of our architecture is to provide real-time decision support in reaction to disturbing events in manufacturing environments. More precisely, we demonstrate how this approach can support engineering change order (ECO) processing in ERP-controlled environments. Our demonstrative example based on a real scenario in the aerospace industry illustrates how a genetic algorithm and a real-time discrete event simulation model can be integrated within an ERP and MES system platform.

**Keywords** – MES, ERP, simulation modelling, genetic algorithm, adaptive manufacturing, integration.

## **B.1 Introduction**

Enterprise Resource Planning (ERP) systems are designed to integrate all internal enterprise business processes. However, ERP systems still show a number of limitations for production management in dynamic and highly variable environments. First of all, ERP inherits two major shortcomings associated with its central MRP planning function: the unlimited capacity resources assumption and its non-stochastic nature (Hopp and Spearman, 2000). Moreover, ERP systems need additional external systems to monitor and collect real-time data. Consequently, the ERP system's inability to handle uncertainties and unexpected events limits its use for supporting decision making processes in dynamic production environment.

This limitation creates additional challenges in organizations producing highly complex products and where the engineering change management effort is important. Complex products such as automotives, aircraft, and major capital equipment and systems sometimes consist of thousands

to hundreds of thousands parts. In addition, there are related to tooling, fixtures, gauges, templates, jigs & dies, test equipment, and software. Each part may undergo ten engineering changes or more over its life which suggests that a company may evaluate and process many thousands of engineering changes for a complex product. Over the life cycle of the product, the manufacturer must assure that the as-designed configuration at any point in time will satisfy functional requirements and that the delivered product corresponds to the approved as-designed configuration.

In order to support the engineering change management process, ERP systems must operate in coordination with the engineering and product development systems. Formal change management ensure that changes to products are captured, authorized, tracked and communicated throughout the enterprise and the entire life cycle of the product. When a change is being evaluated, the following must be considered:

- **Inventory status:** How many items are in inventory? Must they be scrapped or can they be used on other products or modified? What is the cost to modify or scrap?
- **Production status:** How many items are in work-in-process? Can they be reworked to the new configuration considering their current stage of completion, completed and used up before the change is effective? Must they be scrapped? What is the lead-time and cost for production of the new item? What is the additional lead-time for building tooling, fixtures and test equipment?
- **Procurement status:** Is the old item on order? Can it be cancelled and at what cost? What is the lead-time for procuring the new item? Are new suppliers required?
- **Impact on the customers and field service organizations:** What notification is required? How long will the process take? What documentation needs to be updated? What are the implications on spare parts requirements?

In all cases, real-time information is critical to properly evaluate the impact of any design changes. In that context, it is not surprising to note that production departments have always

favoured the development of custom-made production data collection systems using either databases or spreadsheets to monitor and control real-time and variable execution processes (MESA, 1997). Software maintenance and data consolidation is obviously complex in such environment as the number and the structure of these small applications vary over time.

In this paper, we propose a real-time execution system to support engineering change order (ECO) processing in ERP-controlled environments. The main objective of our architecture is to complement the traditional ERP/MES model by providing real-time flexible and well-integrated decision support in reaction to disturbing events in manufacturing environments.

Our paper is structured as follow. We will first present a brief literature review on MES systems, real-time simulation platforms and genetic algorithms used in manufacturing context. The proposed real-time execution architecture will then be presented in Section 3. Section 4 will presents the demonstrative example. A brief discussion of future perspectives will conclude this paper.

## **B.2 Literature review**

For more than 25 years, companies have invested in information systems to achieve productivity gains which brought the enterprise information system market to grow steadily. But for most of that period, the information system specialists did not pay attention to the shop floor (Holst, 2001). In that context, it is not surprising to note that production departments have always favoured the development of custom-made software applications to fill specific necessities as an industrial operations support. Fortunately, the difficulty of integrating multiple point systems has brought software providers to package multiple execution management components into single and integrated solutions. These systems, commonly referred as Manufacturing Execution System (MES), provide a common user interface and data management system. MES functionalities are typically broad and can support the production operations management from point of order release into manufacturing to point of product delivery into finished goods. Using current and accurate data, MES can guide, trigger and reports on plant activities as events occur (MESA,



1997). However, MES applications do not address all of the manufacturing execution processes required to replenish the supply chain while dynamically responding to unpredicted change (Homem-de-Mello et al., 1999, Lobecke and Slawinski, 2004). Reactive manufacturing strategies also require defining the process management layer between ERP and process control and translating demand driven supply chain requirements into a set of capabilities, systems, and workflow integration investments. Therefore, adaptive manufacturing organizations need an execution platform that connect manufacturing processes with engineering and supply chain processes, that enables closed-loop mechanisms, and that provides decision support to production personnel so they can deliver on their performance goals.

Up to now, the scientific literature has mostly address these issues by proposing different integration models between ERP and MES systems (Barry et al., 1998, Hori et al., 1999, Huang, 2002, PABADIS Group, 2001, SEMATECH Inc., 1998, Simao et al., 2006). While these models may resolve some interoperability problems, they fall short of adding decision-making tools capabilities to both systems or to gain insight into manufacturing systems.

In some cases, traditional simulation models may be used off-line to either understand the behaviour of the system or to evaluate various strategies for the operation of the system (Moon and Phatak, 2005). However, simulation models are usually not built for repetitive and real-time usage as the models are not directly coupled with the real information systems (Drake and Smith, 1996). By contrast, on-line decision support systems take a snapshot of the actual factory status and predict future events according to certain modelling assumptions. In a rapidly changing and highly competitive business environment, the decision system therefore needs to have access to critical real-time production information. On-line systems also need to combine different solving tools with multiple data sources to elaborate solutions in reasonable time compatible with the planning horizon.

The advances in computing power and memory over the last decades have opened up the possibility of online simulation based optimization. This recent research development offers one of the most interesting opportunities in simulation and the potential benefits in this field are

significant. A general overview of the different approaches found in the literature including references to the state of the art is provided in (Scott, 2005). A typical area of application for online simulation, also called real-time simulation, is proactive decision support for scheduling problems in manufacturing systems (Siemiatkowski and Przybylski, 2006). Kouiss and Pierreval (1999) present a combination of a MES and an on-line simulation model for controlling a flexible manufacturing system. Simulation is used to evaluate different scenarios proposed by the decision module which takes into account the actual state of the manufacturing system. The selected solution is proposed to a human operator for validation and execution. Shin *et al.* (2004) also discuss a decision problem faced by assembly line personnel when a breakdown occurs in a multiple production line system. A discrete-event simulation based approach is presented to help line personnel to minimize throughput degradation of the line.

In terms of resolution approach, the gradient based search methods and the heuristic methods are the most encountered on-line simulation based optimization approaches (Carson and Maria, 1997). Gradient based search methods cover finite difference estimation, likelihood ratio estimation, perturbation analysis, and frequency domain experiments. These methods aim to estimate the retained performance measure with respect to the decision variables. In the other hand, heuristic methods consist on a random exploration of the admissible solutions in the whole decisions space. The search process ends when the best solution is found. At each point of the search process, the objective function value of the problem is estimated via the simulation model. Thus, no information regarding the analytic form of the objective function is required. This category covers simplex search (Azadivar and Lee, 1998), tabu search, simulated annealing (Lee and Iwata, 1991, Ogbu and Smith, 1990) and genetic algorithms. It is interesting to note that previous researches and survey (Azadivar and Tompkins, 1999, Chaudhry and Luo, 2005, Ruiz et al., 2007) have demonstrated the effectiveness of Genetic algorithms solutions.

In summary, integration between ERP and MES systems is not sufficient to guaranty that organizations will be capable of quickly responding to events as they occur on the shop floor. One of the key differences between off-line simulation model and on-line decision support systems is data. It is why the decision system needs to have access to critical real-time production

information. On-line decision support systems also need to combine different solving tools to be flexible. With those objectives in mind, we propose in the following section a generic manufacturing execution platform where a genetic algorithm integrated with a real-time discrete event simulation models complement the traditional ERP/MES model.

### **B.3 Proposed execution architecture**

It is usually difficult to calculate the impact of a decision on a complex system. In this context, using on-line simulation is attractive but requires the integration of the simulation model and the optimization engine into the enterprise information system landscape and the communication network. It must also combine multiple optimization techniques in order to support different decision-making scenarios.

Our proposed architecture is presented in Figure B.1. The main objective of our architecture is to provide a flexible and integrated real-time decision support in reaction to disturbing events in manufacturing environments. Events can occur from the top level – a new order – or the down level – a breakdown on the shop floor. The different software components are spread over several servers linked by local area networks and/or global network. The connection to the real manufacturing processes is supported by OPC (OLE for Process Control) clients connected to different OPC servers collecting data directly from the process. These clients take in charge raw data acquisition, preliminary storing, and processing. Depending on storing/processing rules, an OPC client can store processed data in an SQL database, exchange the data via XML clients, and generate messages to other components if significant events occur. SQL/XML clients or servers assure data exchange with external software, such as ERP, Web servers, Supervisors etc. They can also generate messages to other system components on the basis of their configuration.

The decision-making nucleus of our system is a hybrid of a discrete-event simulator and optimization tool. The simulator runs models of different parts from the controlled system and responds to incoming messages by initiating decision-making logic depending on the disturbing event and the current system state. At the beginning of the decision making process, the decision-

making nucleus is initialized with data from the ERP (i.e. production order, production sequence, etc.) and from the MES (i.e. order status, machine status, etc.). Simulation is used like a metaheuristic support to evaluate performance indicators under a set of constraints or production rules. In its turn, it sends messages representing decisions made for other actors of the architecture including OPC clients to act on the process controlled.

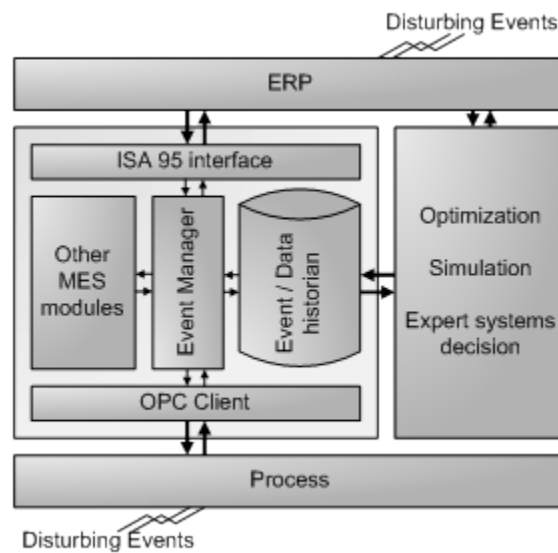


Figure B.1 – Real time execution architecture

## B.4 Case study

In the following example, we illustrate how genetic algorithm and a real-time discrete event simulation model can be integrated within the ERP and MES applications. Readers should note that this demonstrative example is based on a real scenario from the aerospace industry.

### B.4.1 ECO production process

When a design problem is identified by a customer or by an air safety bureau (like the Air Safety Investigation Office in Canada or the National Transportation Safety Board in USA), the equipment manufacturer needs to take corrective actions across multiple departments (engineering, supply, manufacturing, services...) and systems (PLM, ERP, SCM, ...). An

engineering change request (ECR) can impact the product design, the process design, the procurement process, the manufacturing operations, and the in-service operations.

Engineering Change Request has not been intensively studied by the research community and studies done to date have predominantly focused on computer-based tools for analyzing ECR problems, or on qualitative frameworks or methods for implementing ECRs (Wright, 1997). The work on tools has considered topics as updating, disseminating and controlling information, documentation control, and configuration management. Concerning recent works on ECR, Bhuiyan *et al.* (2006) compare the behaviour of management methods to find the most effective one but the model is mainly focused on ECR occurring during new product development. Huang *et al.* (2001) propose a framework to provide better information sharing, simultaneous data access and processing and more prompt communication and feedback and Keller *et al.* (2005) implement algorithm to track and predict change propagation in a complex product context. In the demonstrative example, we address the case where the engineering change request triggers an immediate part replacement and ignore the product and process design revisions. As the part to be replaced can be temporary changed by a similar part, the reader will understand that the scenario is not concerned with safety or regulatory issues.

Once the engineering change request triggers an immediate part replacement action, the manufacturer first needs to check if the item is already in inventory or in work-in-process. If it is not the case, this item has to be produced as soon as possible. In our example, tooling availability is the main constraint and the planner objective is to schedule and release a new production order. The scheduling objective is to minimize the replacement part lead time while minimizing the total makespan.

#### **B.4.2 Systems architecture**

The reactive process is based on the proposed real-time execution architecture. In this scenario, the engineering support team first creates an engineering change request (ECR) in the ERP system. After the completion of a preliminary impact analysis, the ECR is converted into a formal engineering change order (ECO). The reactive scheduling process is then initiated when

the ECO status is updated to a special status asking for a replacement part. The Event Manager automatically sends an alert to the planner who will use the MES to check the inventory and the work-in-process. If needed, a production order will be created in the ERP system. The production planner then determines the tooling and material needed for the newly created production order and then reschedules all orders through the on-line decision support system. In this case, the allotted time period for decision making is sufficient to run an efficient schedule algorithm.

Production orders are released according to this new schedule. When the production is confirmed, an alert is sent to the planner and the replacement part can be shipped to the customer as soon as the good issue transaction is posted.

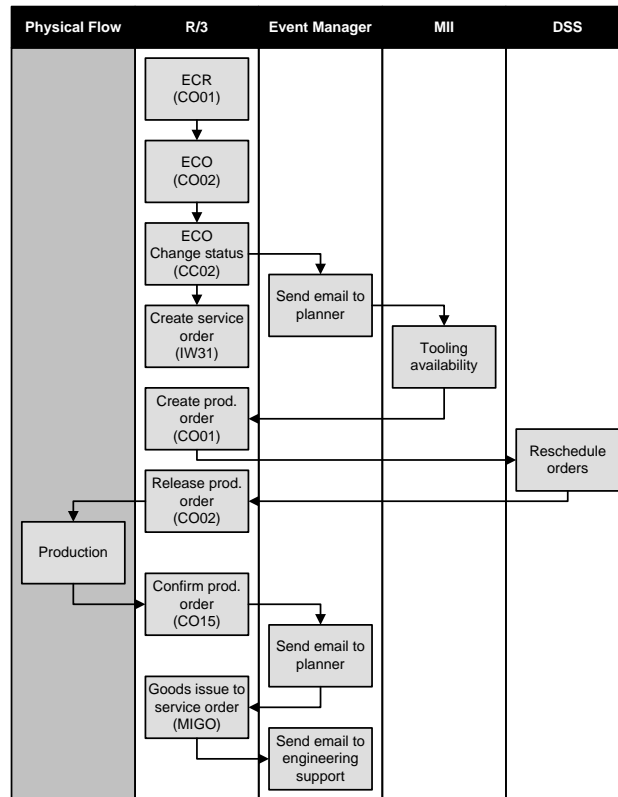


Figure B.2 – ECO scenario system workflow

Figure B.2 presents the decision model process work-flow within the proposed execution system architecture. In this example, the execution system is composed of 3 main components: the ERP system (SAP R/3), the MES system (SAP MII), and the simulation/decision support system (figured by DSS).

### B.4.3 Optimization model

In this scenario, tooling availability is the main constraint. The tooling availability periods are set a long time in advance and cannot be renegotiated. In the study case, 30 jobs are in the system with 3 operations by job, each operation used one of the 4 available tools. Operation  $i$  is always processed on the machine  $i$ .

Processing times and routings are given in Table B.1 and tooling availability periods in Table B.2. ECO  $E^*$  occurs in the system when the job #10 of the initial sequence  $S_0$  begin to be processed. The decision support system tests 3 scenarios (Figure B.3) and the final choice is made by the planner:

- $E^*$  is the first of the to do sequence, the 20 other orders sequence is kept like in  $S_0$  (this solution is actually always used by the company);
- $E^*$  is the first of the to do sequence, the 20 other orders sequence are rescheduled (objective: minimize the total makespan);
- The sequence of the 21 orders is rescheduled (objective: minimize the total makespan).

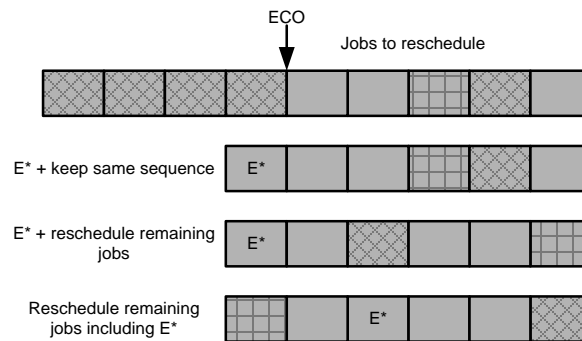


Figure B.3 – ECO tested scenarios

### B.4.3.1 Genetic algorithm

The search process is detailed in Figure B.4. It consists on running the Genetic Algorithm (GA) with respect to its stopping rule and evaluates each desired configuration through the simulation model.

Table B.1 – Processing time and routings

( $d_i$ : operation  $i$  processing time,  $t_i$ : operation  $i$  tooling)

Job	$(d_1, t_1)(d_2, t_2)(d_3, t_3)$	Job	$(d_1, t_1)(d_2, t_2)(d_3, t_3)$
1	(54,3)(34,3)(61,3)	16	(11,3)(14,1)(89,3)
2	(2,4)(9,3)(15,4)	17	(33,3)(62,4)(87,4)
3	(89,3)(70,4)(38,1)	18	(38,4)(79,2)(15,3)
4	(19,1)(28,2)(87,1)	19	(28,4)(30,4)(12,1)
5	(95,1)(34,2)(7,1)	20	(30,1)(69,3)(71,2)
6	(29,3)(79,1)(25,3)	21	(98,4)(74,4)(96,1)
7	(59,2)(95,2)(39,4)	22	(53,2)(80,2)(42,3)
8	(66,1)(89,2)(91,2)	23	(51,1)(14,3)(53,4)
9	(53,4)(1,3)(14,3)	24	(93,2)(43,4)(21,1)
10	(66,1)(50,3)(99,2)	25	(58,3)(12,2)(99,3)
11	(46,3)(11,2)(57,4)	26	(19,4)(57,4)(67,2)
12	(68,3)(67,3)(11,1)	27	(54,1)(83,3)(73,3)
13	(35,1)(51,2)(5,1)	28	(93,4)(50,3)(3,1)
14	(46,1)(27,1)(46,3)	29	(49,3)(70,1)(73,1)
15	(28,1)(40,2)(64,1)	30	(57,1)(21,1)(82,4)

In terms of an optimization problem, the genetic algorithm approach is summarized as follows. At any given point in time, the genetic algorithm generates a population of possible candidate solutions. Initially, the population size is chosen randomly. However this choice typically depends on the characteristics of the problem. Each population component is a string entity of chromosome which represents a possible solution to the problem. The population components are evaluated based on a given fitness function. Highly fit population components are given the



chance to reproduce through a crossover process with other highly fit population elements by exchanging pieces of their genetic information. This process produces «offspring» or new solutions to the optimization problem based upon the high-performance characteristics of the parents. Premature loss of important information by randomly altering bits within a chromosome is prevented by a mutation process. This procedure continues until a satisfactory solution is achieved (Chaudhry and Luo, 2005, Legault, 1994).

Table B.2 – Tooling availability periods

(load / available capacity on 2000 time units = 67%)

Tooling 1		Tooling 2		Tooling 3		Tooling 4	
open	Close	open	close	open	close	open	Close
80	205	0	155	0	55	0	26
245	448	192	439	111	644	39	238
469	773	547	997	656	980	254	502
825	889	1036	1341	1041	1523	547	626
973	1509	1379	1617	1543	1947	660	710
1583	1863	1651	2000			765	896
1939	2000					951	954
						1038	1264
						1301	1727
						1750	1850
						1867	2000

To solve the problem under study, we have developed a GA program based on an existing Toolbox (Chipperfield et al., 1994). The main data structures in the GA toolbox are chromosomes, phenotypes, objective function values and fitness values. The chromosome structure stores an entire population in a single matrix of size  $N_{ind} \times L_{ind}$ , where  $N_{ind}$  is the number of individuals and  $L_{ind}$  is the length of the chromosome structure. Phenotypes are stored in a matrix of dimension  $N_{ind} \times N_{var}$  where  $N_{var}$  is the number of decision variables. A  $N_{ind} \times N_{obj}$  matrix stores the objective function values, where  $N_{obj}$  is the number of objectives. Finally, the

fitness values are stored in a vector of length  $N_{ind}$ . In all of these data structures, each row corresponds to a particular individual.

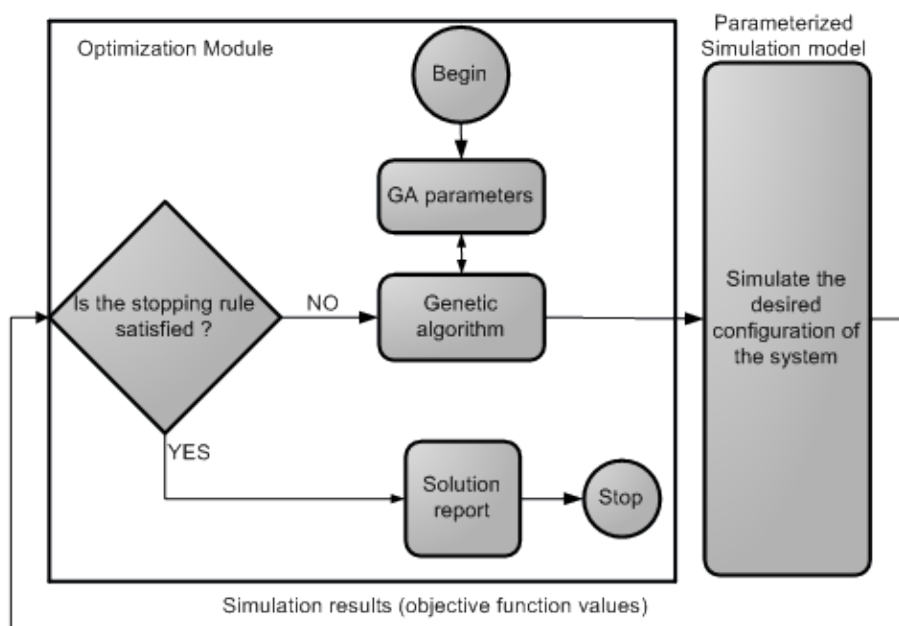
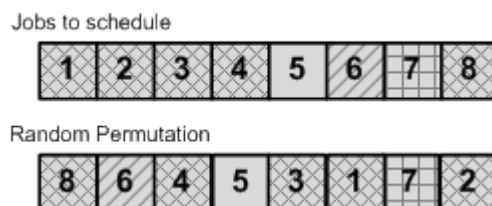


Figure B.4 – Simulation / GA optimization module

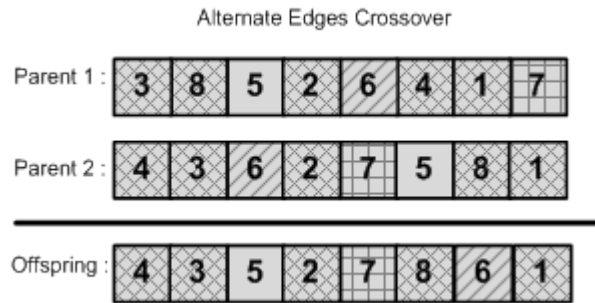
The GA toolbox uses MATLAB matrix functions to build a set of versatile routines for implementing a wide range of genetic algorithm methods. The following steps summarize the employed Genetic Algorithm:

- 1- Population representation and initialisation: permutation coding representation with «  $N_{ind}$  » the number of individuals and  $L_{ind} = 30$  the length of the chromosome.

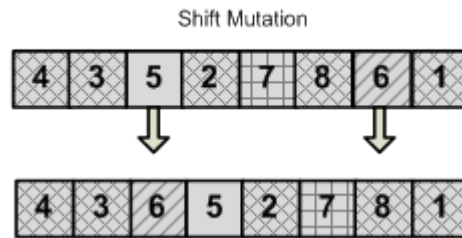


- 2- Fitness: the linear-**ranking** method of Baker (Baker, 1985).
- 3- Selection: roulette wheel selection (Goldberg, 1989) (routine **rws**)

- 4- Crossover: alternate edges crossover (Poon and Carter, 1995) with crossover probability  $P_c$ , a crossover method for ordered chromosomes. It recombines pairs of individuals with given probability to produce offspring with respect to the existing jobs.



- 5- Mutation: shift mutation (Chang et al., 2007) with probability  $P_m$ . A mutation method for ordered chromosomes.



Let « MaxGen » be the maximum number of generation if the stopping algorithm rule is fixed following this criteria.

#### B.4.3.2 Simulation model

The simulation model is build to describe the dynamics of the system governed by the scheduling sequence and the tooling constraints defined previously (section 4.3). These factors are considered as input of such a model and the related total makespan is defined as its output. The parameterized simulation model is developed using the Visual SLAM language (Pritsker et al., 1997).

The Visual SLAM portion is composed of various networks describing specific tasks (tooling constraints, sequence, data exchange with Genetic algorithm, etc...). The simulation ends when

we reach the end of the defined production sequence. Figure B.5 shows a bloc diagram representation of the simulation model.

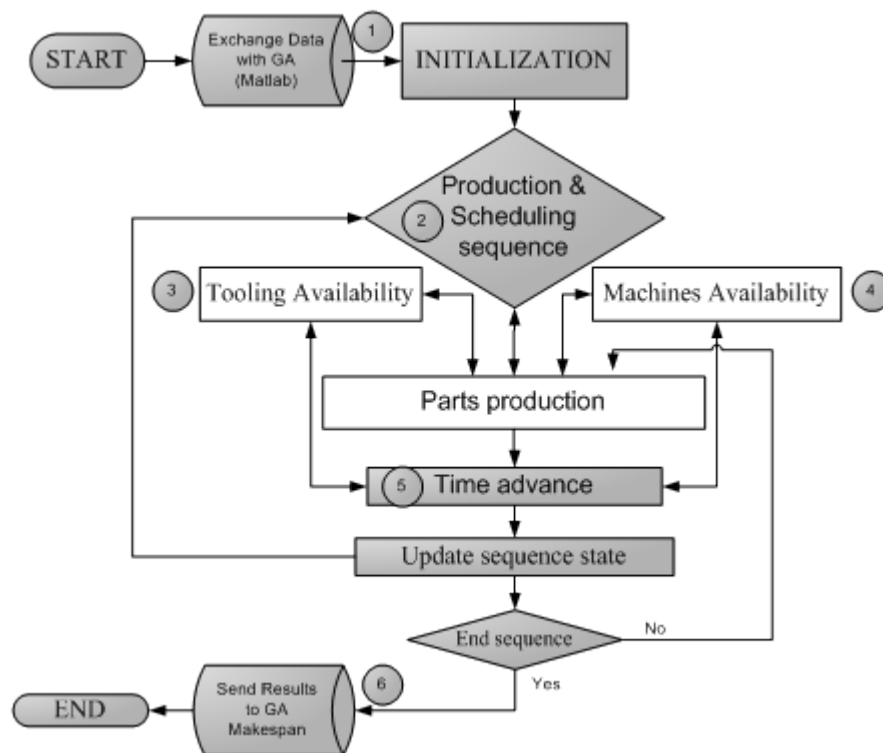


Figure B.5 – Simulation model bloc diagram

1) The Exchange data block read the parameters of each individual of the population set by the Genetic Algorithm. To run the model the INITIALIZATION block sets these values and other parameters defining the system (e.g., the tooling constraints, the machines availabilities, the processing times and routings,...) as well as the simulation stopping criteria.

2) The production and scheduling sequence mechanism set the processing time and the part type to produce according to the generated sequence. This block is in connection with the «update sequence state» block charged to send a signal when a given part reaches the end of the process. The scheduling sequence is then activated to launch the following part.

- 3) The tooling availability bloc set the availability periods of tooling 1 to 4 according to the cycle defined in Table B.2 and provided by the MES. This bloc is in connection with the « parts production » and the « time advance » blocs. In fact, when a tooling is being used it becomes unavailable for other jobs. Moreover, when the close time defined by Table B.2 is reached the tooling should be preempted.
- 4) The machines availability block samples the times to failure and times to repair for the machines from their respective probability distributions. This bloc is in connection with the « parts production » and the « time advance » blocs. In fact, when a machine is being repaired it becomes unavailable for other jobs. According to the time advance the available / unavailable states are defined by a two well defined signals.
- 5) The time advance block follows the system events progress and governs the tooling and machines availabilities blocks.
- 6) The Send Results block writes the incurred total makespan of each individual of the genetic algorithm population in an external file. This file being available to the genetic algorithm program reiterates the optimization process.

#### **B.4.4 Validation**

It is extremely difficult to compare the performance of different evolutionary algorithms since most researchers use their own instances of test problems, i.e., problems where the processing times and due dates of the jobs are selected randomly out of a uniform distribution. To validate our model, we use a standard benchmark problem taken from Taillard (1993). This flowshop instance has 20 jobs and 5 machines, uses the time seed 379008056. A calculated lower bound is 1290 and the best makespan found by Taillard is 1359. With our model we found 1360 after 100 iterations and with  $P_c$  equal to 0.6. Being very close to Taillard results we decided to go further with the developed genetic algorithm and the problem under study. It is interesting to note that when dealing with genetic algorithms, the choice of the GA parameters (i.e.,  $P_c$ ,  $P_m$ ,...) is an

important issue to be taken into account since it can affect the optimization process and the final results. In the research literature the choice of these parameters is generally based on experience.

#### **B.4.5 Results – discussion – extension**

The system parameters and ECO data used to run the optimization module and to characterize the best scheduling / rescheduling scenario are given in Table B.1, Table B.2 and Table B.3, respectively.

Table B.3 – ECO Processing time and routing

Job	$(d_1, t_1)(d_2, t_2)(d_3, t_3)$
ECO	$(57, 1)(21, 1)(82, 4)$

The obtained results are given in Figure B.6. When the same sequence before ECO is kept, the insertion of  $E^*$  in the first position leads to a total makespan of 3621 for the 21 orders. This solution is actually used by the company. When the decision to reschedule the remaining jobs including ECO is taken, the total makespan is reduced to up 13.6 %. It is interesting to note that under the considered case and with respect to the optimization objective, the difference between the second and the third scenario (see Figure B.6) is not significant. In the next section, it will be shown that the gap between the two results will be amplified under more constrained configurations.

To illustrate the effect that some considered system parameters variation have on the solutions, a sensitivity analysis was conducted. Table B.4, Table B.5 and Table B.7 detail the considered parameters variations, and present the obtained total makespan for the sensitivity analysis cases.

Due to the number of the involved parameters we decided to limit our analysis to the tooling constraints and the ECO parameters. These parameters could be considered as illustrative in our context. Moreover, our objective is to insure the robustness of the approach and the proposed solutions. We claim that, at this point, this objective is reached and it will be reinforced with this sensitivity analysis.

The obtained results are shown in Table B.6 and Table B.8 for the second and third run respectively. The first line of these tables concerns the total makespan for the initial 30 jobs. The following lines are about the three tested scenarios (Figure B.3). Following our expectations, the results show that facing a more constrained system (i.e., tooling availability and ECO routing) the total makespan is increasing. Moreover, the gap between the actual practices (i.e., insert ECO at the beginning and keep the same schedule) and the two other solutions is increasing. These results point toward a considerable benefit for the company.

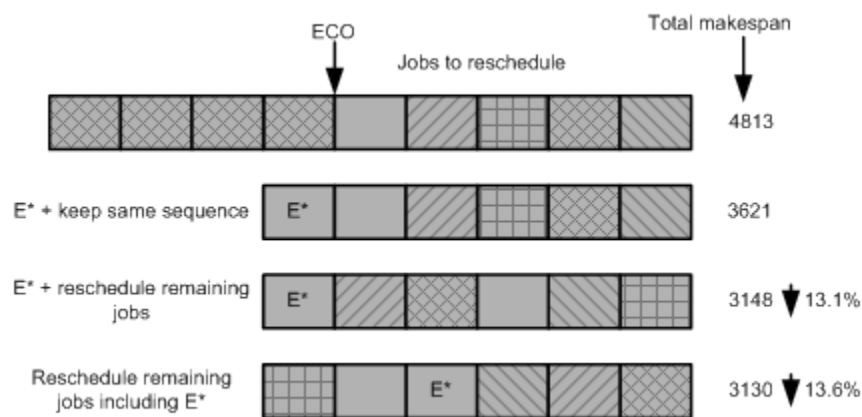


Figure B.6 – Obtained results

Table B.4 – Tooling availability periods for the 2nd run

(load / available capacity on 2000 time units = 77%)

Tooling 1		Tooling 2		Tooling 3		Tooling 4	
open	Close	open	close	open	close	open	close
80	152	0	114	13	158	0	96
192	309	177	287	174	355	158	162
330	504	335	435	400	458	211	217
556	593	518	804	492	528	313	501
677	983	813	987	583	678	512	672
1057	1218	1079	1124	733	735	728	886
1294	1630	1180	1367	819	984	983	1086
1657	1772	1450	1519	1021	1332	1121	1172
1817	2000	1589	1840	1355	1428	1255	1496
		1907	2000	1445	1808	1560	1756
				1828	2000	1856	1860
						1927	2000

Table B.5 – ECO Processing time and routing

Run	Job	$(d_1, t_1)(d_2, t_2)(d_3, t_3)$
2 <sup>nd</sup>	ECO	(47,1)(53,3)(44,4)
3 <sup>rd</sup>	ECO	(47,3)(53,4)(44,1)

Table B.6 – Obtained results for the 2nd run

	Total makespan	Variation
30 jobs sequence	5647	
E* + keep same sequence	4660	
E* + reschedule remaining jobs	3984	↓ 14,51%
Reschedule remaining jobs including E*	3913	↓ 16,05%



Table B.7 – Tooling availability periods for the 3rd run

(load / available capacity on 2000 time units = 96%)

Tooling 1		Tooling 2		Tooling 3		Tooling 4	
open	Close	open	close	open	close	open	close
155	227	67	211	0	222	165	294
304	421	405	663	371	445	366	610
462	636	733	909	570	872	655	713
737	774	977	1115	992	1011	746	1030
936	1242	1176	1405	1037	1182	1069	1218
1386	1547	1593	1628	1214	1395	1322	1418
1694	2000	1740	1993	1485	1543	1540	1544
				1610	1646	1640	1646
				1754	1849	1834	2000
				1958	2000		

Table B.8 – Obtained results for the 3d run

	Total makespan	Variation
30 jobs sequence	7044	
E* + keep same sequence	6485	
E* + reschedule remaining jobs	4730	↓ 27,06%
Reschedule remaining jobs including E*	4537	↓ 30,04%

In this context, to minimize the total makespan, it is always more interesting to reschedule the remaining jobs including E\*. However, based on the due date of the ECO the planner should make a multi objective choice (i.e., minimize total makespan and deliver ECO as soon as possible while respecting the deadline). But if the last strategy proposes a schedule with the ECO in late, this solution will be eliminated by the planner even if it is a better choice in term of makespan. If

we want the GA proposes a new schedule under the ECO due date constraint, we should update the cost function to integrate the ECO tardiness. Equation (1) combines the actual makespan with the weighted quadratic tardiness. In one hand, if all the weights  $\omega_i$  are set to 0 for the non-ECO jobs, the obtained result can be better than the one with the ECO at the first place and the remaining jobs reschedule. In the other hand, the weights can be used to model the fact that some jobs are made to order and others are made to stock. The tooling shop schedule is synchronized with the assembly schedule and with the cost function (1) we will be able to propose schedules more respectful of the global problem.

$$F' = \alpha \times \left( \omega_{ECO} T_{ECO}^2 + \sum_{i \in J} \omega_i T_i^2 \right) + (1 - \alpha) \times makespan \quad (1)$$

with:  $T_i = \max(0, C_i - D_i)$

$C_i$  : completion time of job  $i$

$D_i$  : due date job  $i$

For  $i$  in  $J$ ,  $J$ : the set of jobs to schedule.

#### B.4.6 Optimization Process performance

As we propose an executive architecture, it is important to measure the performance of the optimization process integrated in the ERP-controlled environment. This performance can be expressed as the time used by the process to return a solution and can be decomposed as an ERP time, a simulation time and a CPU time. The ERP time is the time spent to retrieve the required data to initialize the simulation model and the GA, to send the data to and receive the results from the optimization nucleus and to update the start date of orders. This time can fluctuate from 30 to 45 seconds depending on the network and the ERP server load. The simulation time is the time used by the optimization nucleus to load the sequence sent by the GA, simulate each sequence and send the results to the GA. The CPU time is the time used by the GA to generate new sequences. The CPU and simulation times are related and depend on the number of sequences

generate at each step of the GA (i.e.,  $N_{ind}$ ) and the number of steps the GA processes (i.e., MaxGen) to provide a solution. The Figure B.7 provides a computation time analysis of the problem using the tooling calendars presented in the Table B.7. The more complex calendars are used for this analysis because we note the simulation time depends on the calendars complexity and can be reduce by 20 seconds for easier calendars.

As shown by Figure B.7, The GA parameters  $N_{ind}$  and MaxGen have an impact on the main part of the time spent to solve the problem and it is important to find the better sets for each plant depending on the size of the problem and planner needs. Keeping it in mind, the time performance can fluctuate a lot but it is always acceptable (from 1.5 to 6 minutes). Obviously, one should note the significant impact that the GA parameters have on the solution (i.e., minimize makespan). Hence, the same sensitivity analysis is conducted to study this issue. As shown by Figure B.8 and Table B.9 the best values of the Makespan are reached and stabilized around  $N_{ind} = 120$  and MaxGen = 120. To join the aforementioned idea, it is important to find a compromise between the time spent to solve the problem and the planner needs. In fact, between a Makespan equal to 7044 and reached after 6 minutes and the best makespan equal to 7042 and reached after 10 minutes one should make a choice.

Table B.9 – Comparative study

Cases	MaxGen	Nind	Execution Time	Makespan	Execution Time Variation	Makespan Variation
Basic	50	50	92,563	7494	---	---
2	100	100	351,02	7044	↑279 %	↓6,00 %
3	120	120	484,24	7042	↑423 %	↓6,03 %
4	130	130	602,36	7042	↑551 %	↓6,03 %

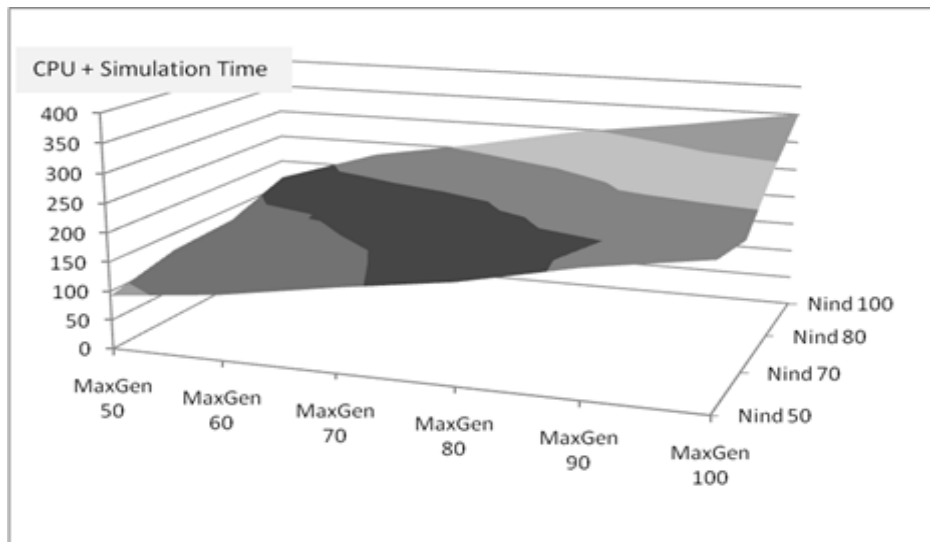


Figure B.7 – Execution time evolution

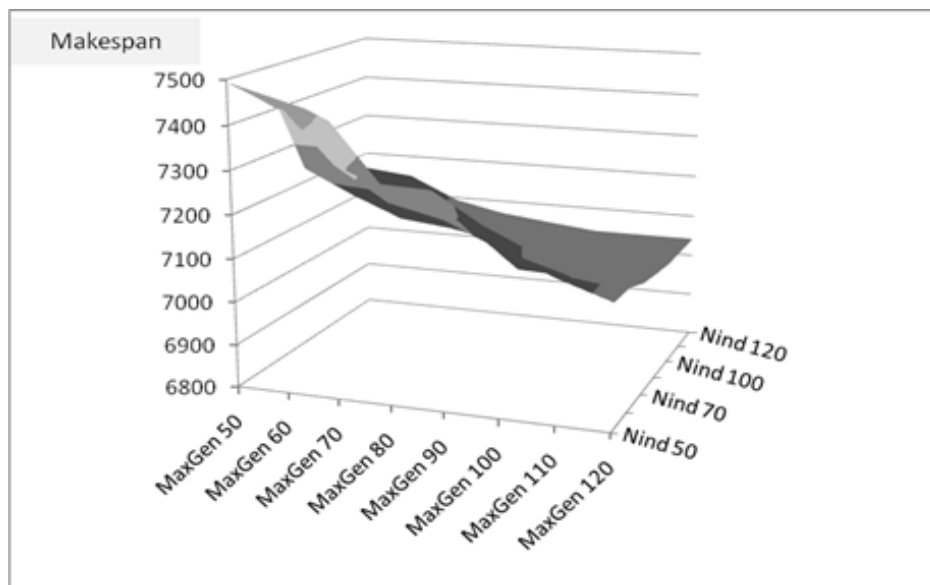


Figure B.8 – Makespan evolution

## B.5 Conclusion

This paper presents our contribution to the development of an open architecture framework supporting real-time decision-making. The proposed real-time control system is integrating optimisation and simulation methods with ERP and MES systems. The functionalities and the interactions between the different components of the real-time execution system were presented. A demonstrative example also illustrated the operating mode of the functional architecture of the control system. This real life based application highlighted the integration aspects of the different control system modules and their usefulness to solve real-time manufacturing decision-making problems. Other scenarios will be studied in the future to demonstrate the generic aspect of our approach in solving real-time decision-making problems of adaptive manufacturing systems. Future works will also involve the integration of other optimisation techniques.

It is interesting to note that we have only considered a production shop floor without external constraint. In fact, the original case consists on a production shop floor which supplies an assembly line. This configuration set additional constraints on the production shop floor schedule. In this context, the production shop floor produces items on stock and on demand. When an ECO occurs, the planner has to reschedule the shop floor with a minimum impact on the assembly line schedule while always minimizing the total makespan. This case study will be investigated in further research.

## B.6 References

- Azadivar, F. & Lee, Y.-H. (1998) Optimization of discrete variable stochastic systems by computer simulation. *Mathematics and Computers in Simulation*, 30, 331-345.
- Azadivar, F. & Tompkins, G. (1999) Simulation optimization with qualitative variables and structural model changes: A genetic algorithm approach. *European Journal of Operational Research*, 113, 169-182.

- Baker, J. E. (1985) Adaptive selection methods for genetic algorithms. *Proceedings of the 1st International Conference on Genetic Algorithms*.
- Barry, J., Aparicio, M., Durniak, T., Herman, P., Karuturi, J., Woods, C., Gilman, C., Lam, H. & Ramnath, R. (1998) NIIP-SMART: an investigation of distributed object approaches to support MES development and deployment in a virtual enterprise. *Proceedings of Second International Enterprise Distributed Object Computing Workshop, 1998. EDOC '98*.
- Bhuiyan, N., Gregory, G. & Vince, T. (2006) Engineering change request management in a new product development process. *European Journal of Innovation Management*, 9, 5-19.
- Carson, Y. & Maria, A. (1997) Simulation optimization: Methods and applications. *Proceedings of the 1997 Winter Simulation Conference*. Atlanta, GA, IEEE Piscataway NJ USA.
- Chang, P.-C., Hsieh, J.-C. & Wang, C.-Y. (2007) Adaptive multi-objective genetic algorithms for scheduling of drilling operation in printed circuit board industry. *Applied Soft Computing Journal*, 7, 800-806.
- Chaudhry, S. S. & Luo, W. (2005) Application of genetic algorithms in production and operations management: A review. *International Journal of Production Research*, 43, 4083-4101.
- Chipperfield, A. J., Fleming, P. J. & Fonseca, C. M. (1994) Genetic algorithm tools for control systems engineering. *Proceedings of Adaptive Computing in Engineering Design and Control*. Plymouth Engineering Design Center (UK).
- Drake, G. R. & Smith, J. S. (1996) Simulation system for real-time planning, scheduling, and control. *1996 Winter Simulation Conference*. Coronado, CA, USA, IEEE, Piscataway, NJ, USA.
- Goldberg, D. E. (1989) *Genetic Algorithms in Search, Optimization and Machine Learning*, Boston, MA, USA, Addison-Wesley.
- Holst, L. (2001) Integrating Discrete-Event Simulation into the Manufacturing System Development Process. Lund, Sweden, Lund University, Department of Mechanical Engineering, Division of Robotics.

Homem-De-Mello, T., Shapiro, A. & Spearman, M. L. (1999) Finding optimal material release times using simulation-based optimization. *Management Science*, 45, 86-101.

Hopp, W. & Spearman, M. (2000) *Factory Physics*, New York , NY, McGraw-Hill.

Hori, M., Kawamura, T. & Okano, A. (1999) OpenMES: scalable manufacturing execution framework based on distributed object computing. *1999 IEEE International Conference on Systems, Man, and Cybernetics 'Human Communication and Cybernetics'*. Tokyo, Jpn, Institute of Electrical and Electronics Engineers Inc. USA.

Huang, C.-Y. (2002) Distributed manufacturing execution systems: A workflow perspective. *Journal of Intelligent Manufacturing*, 13, 485-497.

Huang, G. Q., Yee, W. Y. & Mak, K. L. (2001) Development of a web-based system for engineering change management. *Robotics and Computer-Integrated Manufacturing*, 17, 255-267.

Keller, R., Eckert, C. M. & Clarkson, P. J. (2005) Multiple views to support engineering change management for complex products. *3rd International Conference on Coordinated and Multiple Views in Exploratory Visualization, CMV 2005*. London, United Kingdom, Institute of Electrical and Electronics Engineers Computer Society, Piscataway, NJ 08855-1331, United States.

Kouiss, K. & Pierreval, H. (1999) Implementing an on-line simulation in a flexible manufacturing system. *Simulation in Industry'99. 11th European Simulation Symposium 1999. ESS'99*. Erlangen, Germany, SCS.

Lee, Y.-H. & Iwata, K. (1991) Part ordering through simulation-optimization in an FMS. *International Journal of Production Research*, 29, 1309-1323.

Legault, G. (1994) Un algorithme génétique pour la conception topologique de réseaux téléinformatiques à commutation de paquets. Université du Québec à Montréal.

Lobecke, R. & Slawinski, T. (2004) Integrated manufacturing execution systems - history and current state: from system integration to overall optimisation of business processes. *Automatisierungstechnische Praxis*, 46, 21-5.

- Mesa (1997) White Paper #06: MES Explained: A High Level Vision for Executives. Pittsburg (PA), USA, MESA International.
- Moon, Y. B. & Phatak, D. (2005) Enhancing ERP system's functionality with discrete event simulation. *Industrial Management and Data Systems*, 105, 1206-1224.
- Ogbu, F. A. & Smith, D. K. (1990) Application of the simulated annealing algorithm to the solution of the  $n/m/C_{\max}$  flowshop problem. *Computers & Operations Research*, 17, 243-253.
- Pabadis Group (2001) Revolutionizing Plant Automation : the PABADIS Approach. Deliverable 6.3.
- Poon, P. W. & Carter, J. N. (1995) Genetic algorithm crossover operators for ordering applications. *Computers & Operations Research*, 22, 135-147.
- Pritsker, A. A. B., O'reilly, J. J. & Laval, D. K. (1997) *Simulation with visual SLAM and AweSim*, New York (NY), John Wiley & Sons, Inc.
- Ruiz, R., Carlos Garcia-Diaz, J. & Maroto, C. (2007) Considering scheduling and preventive maintenance in the flowshop sequencing problem. *Computers and Operations Research*, 34, 3314-3330.
- Scott, D. (2005) Comparative advantage through manufacturing execution, and applications. IN INGALLS, R. G., ROSSETTI, M. D., SMITH, J. S. & PETERS, B. A. (Eds.) *Proceedings of the 2005 Winter Simulation Conference*.
- Sematech Inc. (1998) *Computer Integrated Manufacturing Framework Specification Version 2.0*, Austin, TX.
- Shin, F., Ram, B., Gupta, A., Yu, X. & Menassa, R. (2004) A decision tool for assembly line breakdown action. *2004 Winter Simulation Conference*. Washington, DC, United States, Institute of Electrical and Electronics Engineers Inc., New York, NY 10016-5997, United States.
- Siemiatkowski, M. & Przybylski, W. (2006) Simulation studies of process flow with in-line part inspection in machining cells. *Journal of Materials Processing Technology*, 171, 27-34.



Simao, J. M., Stadzisz, P. C. & Morel, G. (2006) Manufacturing execution systems for customized production. *Journal of Materials Processing Technology*, 179, 268-275.

Taillard, E. (1993) Benchmarks for basic scheduling problems. *European Journal of Operational Research*, 64, 278-285.

Wright, I. C. (1997) A review of research into engineering change management: Implications for product design. *Design Studies*, 18, 33-39.

## **ANNEXE C: DEVELOPMENT AND INTEGRATION OF A REACTIVE REAL-TIME DECISION SUPPORT SYSTEM IN THE ALUMINUM INDUSTRY**

Référence : Saenz de Ugarte, B., Hajji, A., Pellerin, R., & Artiba, A. (2009). Development and integration of a reactive real-time decision support system in the aluminum industry. *Engineering Applications of Artificial Intelligence*, 22(6), 897-905.

**Abstract** – This paper aims at providing real-time decision support in reaction to disruptive events in manufacturing environments. More precisely, we demonstrate how this approach can support the rescheduling process in ERP-controlled environments. Our demonstrative example, based on a real scenario in the aluminum industry, illustrates how a genetic algorithm and a real-time discrete event simulation model can be integrated within common enterprise information systems.

**Keywords** – Rescheduling, genetic algorithm, real-time decision, MES, ERP.

## C.1 Introduction

As complexity and demand for new or customized products grow, companies are realizing that their supply chains need to be more flexible and responsive. Responsiveness can be achieved in different ways including exploiting flexibility in supply chain structures (Wadhwa *et al.*, 2008), by reducing cycle time, and by implementing a customer-driven replenishment process. A responsive enterprise can eventually survive competition by operating as a effective member of a supply chain network and by its ability to make a rapid and balanced response to predictable and unpredictable change (Ranga and Dwivedi, 2003).

As reaction time decreases however, shop floors become more complex, dynamic and consequently unstable. In dynamic supply chain networks, manufacturers have to react to various unforeseen events occurring within or outside the company boundaries (Saad and Gindy, 1998) which draws attention to the importance of tracking material flow and information flows and of making decision quickly. The responsive system therefore needs to have access to critical real-time production information and combine different solving tools with multiple data sources to elaborate solutions in reasonable time compatible with the planning horizon.

In the supply chain literature, traditional simulation models have been used extensively for many years as strategic decision tools to either understand the system behaviour or to evaluate off-line various strategies for the system operation. However, simulation models are usually not

employed for repetitive and real-time usage as the models are not directly coupled with the real information systems.

By contrast, online decision support systems take a snapshot of the current factory status and predict future events according to certain modeling assumptions. In a rapidly changing and highly competitive business environment, the decision system therefore needs to have access to critical real-time production information. A typical area of application for online simulation, also called real-time simulation, is proactive decision support for scheduling problems in manufacturing systems. For example, Kouiss and Pierreval (1999) present a combination of a Manufacturing Execution System (MES) and an online simulation model for controlling a flexible manufacturing system. Simulation is used to evaluate different scenarios proposed by the decision module, which takes into account the current state of the manufacturing system. The selected solution is proposed to a human operator for validation and execution. Shin *et al.* (2004) also discuss a decision problem faced by assembly line personnel when a breakdown occurs in a multiple production line system. A discrete-event simulation based approach is presented to help line personnel minimize throughput degradation of the line.

Using online simulation is attractive but requires the integration of the simulation model into the enterprise information system landscape and the communication network. It must also combine multiple optimization techniques in order to support different decision-making scenarios. Fortunately, the advances in computing power and memory over the last decades have opened up the possibility of online simulation based optimization. This recent research development offers one of the most interesting opportunities in simulation and the potential benefits in this field are significant. In terms of a resolution approach, gradient based search methods and heuristic methods are the most encountered online simulation based optimization approaches (Carson and Maria, 1997). Gradient-based search methods cover finite difference estimation, likelihood ratio estimation, perturbation analysis and frequency domain experiments. These methods aim to estimate the retained performance measure with respect to decision variables. On the other hand, heuristic methods consist of a random exploration of the admissible solutions in the whole space of the decision. The search process ends when the stop condition is filled. At each point in the

search process, the objective function value of the problem is estimated via the simulation model. Thus, no information regarding the analytic form of the objective function is required. This category covers simplex search, tabu search, simulated annealing and genetic algorithms (Azadivar and Lee, 1998; Carson and Maria, 1997; Chaudhry and Luo, 2005; Ruiz *et al.*, 2007).

In this paper, we propose a generic manufacturing execution architecture in which a genetic algorithm integrated with a real-time discrete event simulation model complements the traditional information system landscape used by manufacturing organizations. The main objective of our architecture is to provide real-time decision support in reaction to disruptive events in manufacturing environments. The remainder of this paper is structured as follows. Section 2 presents the demonstrative example based on a real scenario in the aluminum industry. Section 3 presents our resolution approach. The integration of the proposed approach within an Enterprise Resource Planning (ERP) and MES platform is then presented in Section 4. Finally, a brief discussion of future perspectives will conclude this paper.

## **C.2 Case problem**

The aluminum transformation process is a hybrid continuous and discrete manufacturing process. In the early stage of the process, anodes made of oil coke and tar are bathed in a hot electrolyte. The electrolyte then dissolves the alumina and the aluminum is collected in pots. The chemical reaction on anodes takes many days (continuous process). When a pot is full, it is then siphoned in a crucible (discrete process). The crucible is used to transfer the melting aluminum from pot lines to a furnace. In the furnace, alloys are prepared and mixed. At the end of the process, the alloy is transferred to the cast line to make alloy bars.

In our example, a pot line has a capacity of 240 pots. Pots are tapped individually every 24 hours. Each individual pot has chemical properties and a scheduling time. Chemical properties can change over time and are updated with lab values. Chemical properties include the level of metal contamination. Depending on contamination levels, some pots may not be used for producing high quality alloys (higher market value).

When the metal is put into crucibles, it is transported to the cast lines. Metal arrives in the cast house in an uncontrolled sequence which depends on the real tapping schedule.

Batches of alloys are produced in the cast lines. As metal always arrives in a non-controlled sequence, crucibles may need to be emptied into a waiting furnace. The chemical properties of the waiting furnace therefore vary according to the arrival and properties of new crucibles.

The operator of this manufacturing system aims at different objectives. At the corporate level, the objective is to optimize the allocation of alloys between plants according to the conditions and capacity of each plant. This problem is not the concern of this paper. Our focus will be rather on the plant level, where the objective is to optimize the daily schedule according to the real-time shop floor conditions. The latter objective is measured by comparing the quality of the metal produced to the quality sold to the customer. Globally, the objective of the shop floor personnel is to restrict the use of high quality metal for customer orders of a higher valued alloy and minimize the use of high quality metal for customer orders requesting low alloy grades.

Plant allocation of alloys and production orders are currently managed by the foundry ERP. The scheduler uses no particular tools to produce the detailed schedule. Orders are released in the ERP and the weekly schedule is sent to the shop floor. A MES system is used to track the production status and the aluminum quality at the earliest stage of the manufacturing process. When an order is partially or totally complete, the order status is updated in the ERP. When a new order is allocated to the plant, or when an order is partially completed (because of a lack of aluminum with a sufficiently high), the scheduler has to update the schedule that is made twice a week. Actually, decisions are made by the scheduler. Thus, the quality of the decisions depends strongly on his experience and the time available to make the schedule. This fact creates an important variance from one scheduler to another.

This reality based application highlights the integration aspects of the different control systems and the need to complement them with real-time decision-making capabilities.

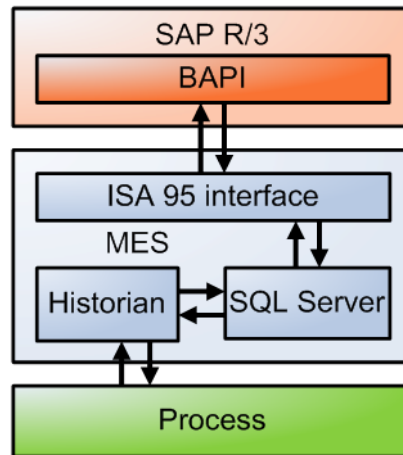


Figure C.1 – System environment

## C.3 Resolution approach

As mentioned in the previous section, the existing ERP-controlled environment is insufficient in guaranteeing a quick response to random events as they occur on the shop floor. Moreover, it falls short of meeting the quality objectives of the company. To bring a feasible solution that can be easily integrated into the existing architecture illustrated in Figure C.1, there is a need to complement it with real-time decision-making capabilities. This section details the proposed architecture and highlights the interactions between the different components.

### C.3.1 Proposed execution architecture

Our proposed architecture is presented in Figure C.2. The main objectives of our architecture are to provide real-time decision support in reaction to disruptive events in manufacturing environments and an optimization capability for daily scheduling decisions. Events can occur from the top level: a new order, or the bottom level: a breakdown on the shop floor. The different software components are spread over several servers linked by local area networks and/or a global network. The connection to the real manufacturing processes is supported by the historian connected to different OPC (OLE for Process Control) servers collecting data directly from the process. This historian takes charge of raw data acquisition, preliminary storing, and processing.

The SQL server takes care of manual entry in the MES and, depending on storing/processing rules, it can return raw data from historian archives, process more complicated calculations, and trigger store procedures or generate alarms to other components if significant events occur. It also ensures data exchange with external software, such as ERP, Web servers, Supervisors, and more.

The SAP xApp Manufacturing Integration and Intelligence (xMII) provides a direct connection between shop floor systems and business operations. It ensures that all relevant data is visible in real time, including information about orders, materials, equipment status, costs, and product quality. xMII provides a rich set of integration elements; pre-built, standards-compliant connectors between the shop floor and enterprise systems significantly reduce the development complexity. Organizations can freely combine those components to create composite applications fusing manufacturing execution and process with enterprise workflows. For our part, we use xMII to simplify interactions between the different components of the execution architecture. The JCo connector is thus used to communicate with SAP R/3, OCDB connector to dialog with the MES and with SDBuilder, and flat files generator to interact with the genetic algorithm.

In our case, the xMII makes it possible to launch the decision-making nucleus of our system via the two events occurring either from the top level (i.e., daily scheduling) or from the bottom level (i.e., a breakdown on the shop floor). These two events are illustrated in Figure C.2 by arrow IE1 and IE2, respectively. Interestingly, the reaction to the disruptive event is made via an intelligent connection between the xMII and the simulation model.

The decision-making nucleus of our system is the combination of a genetic algorithm and a discrete event simulator. The simulator runs models of different parts from the controlled system. The genetic algorithm responds to incoming messages by initiating decision-making logic, depending on the disruptive event and the current system state. In what follows, we explain how the two previously mentioned events initiate and launch the decision making process or suspend it to update the process with real time data.

- At the beginning of the decision making process, the two components are initialized through xMII with data from the ERP (i.e. production order, production sequence, etc.)



and from the MES (i.e. order status, machine status, etc.). In turn, the simulator sends messages representing the decisions made. Information is only shared through one channel and xMII is used as a hub to provide required data to all the components of the architecture. Finally, when the genetic algorithm is converging, the best solutions are sent back to xMII (illustrated by arrow R in Figure C.2). The user can choose the best solution, or adapt it to details he can't share with the systems or that are not incorporated into the decision model. The user interface can also be used solely to inform the end user and show him the new schedule.

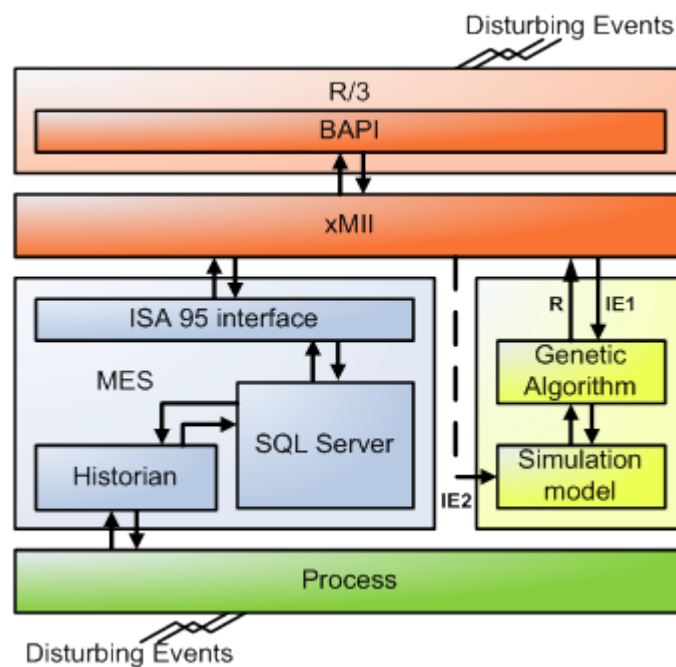


Figure C.2 – Real time execution architecture

- During the decision making process, the occurrence of a disruptive event (i.e., a furnace breakdown) suspends the optimization process to update the system with the new state of the shop floor. The main feature at this point is to provide the decision-making nucleus with a simulator that is able to access real-time shop-floor data for production control.

The SD Builder is connected to the real world by means of a message system. This allows it to have online data about the real system state, to react to events occurring in the real system and to act on the system. Maintaining the current model of the system being controlled, the SD Builder runs models of different parts of the system and responds to incoming messages by initiating, if necessary, a decision-making logic, depending on the message received and the current system state. While making a decision, it can run simulation models starting from the current state of the system controlled, trigger an expert system inference process from the current state of the database and initiate the optimization using the genetic algorithm. We refer the reader to Iassinovski et al. (2008) to obtain more details on SD Builder capabilities.

### C.3.2 Data exchange and interactions

The decision process is based on the proposed real-time execution architecture presented in Figure C.2. In this scenario, the master planner dispatches orders to foundries. When the foundry receives an order, the reactive scheduling process is then initiated. The shop floor status is updated as well as order status at the casting end.

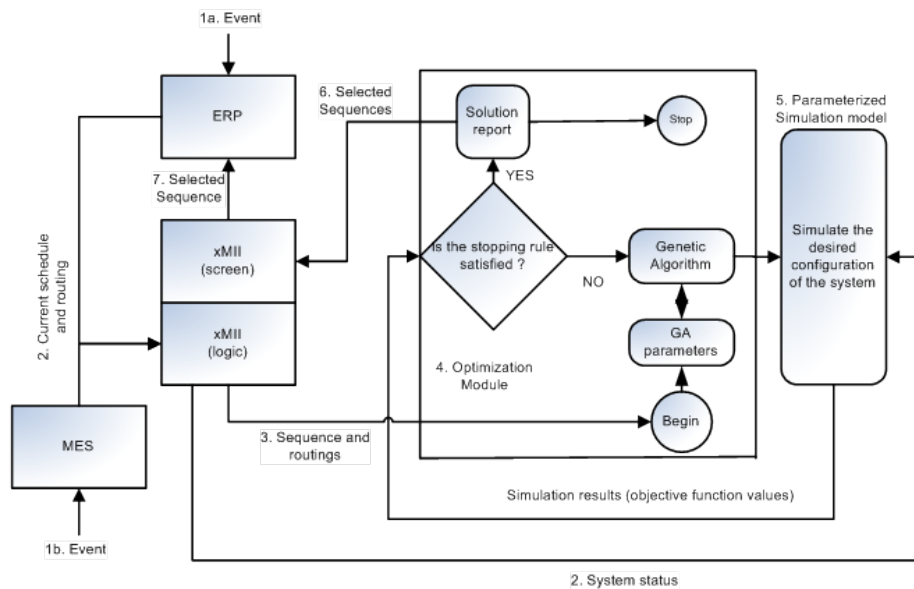


Figure C.3 – Simulation / GA optimization module logic

The search process is detailed in Figure C.3. It consists of running the genetic algorithm with respect to its stopping rule and evaluates each desired configuration through the simulation model. Two event occurrences drive the schedule update process: a new order arrival or the occurrence of a disruptive event. These events are illustrated in Figure C.3 by arrows 1a and 1b, respectively.

## C.4 Implementation of the proposed model

In this section, the implementation issue of the real time execution architecture is presented. The optimization module elaboration as well as its interaction with the integration component (xMII) is primarily discussed.

### C.4.1 Optimization model

In this scenario, mixing furnaces are the main constraint. While it is the bottleneck of the smelter, the planner only generates one production sequence for all of the furnaces. When a mixing furnace is free, it will process the first non-allocated order of the sequence. In the case study, 20 jobs are in the system and two of them are currently processed by the mixing furnaces. Quantities and product details are given in Table C.1. The maximum load of a furnace is 24 tons, and there are three kinds of alloys ( $n = 3$ ).

The main objective of the optimization is to generate a sequence minimizing the use of high quality metal  $i$  for customer orders requesting low alloy grades  $j$ . The cost function used by the optimization model is as follows:

$$\min \sum_i \sum_j \omega_{i,j} \times q_{Di,j} \quad (1)$$

$$\omega_{i,j} = \begin{cases} (j-i)(n-i+1) & \text{if } j > i \\ 0 & \text{else} \end{cases} \quad (2)$$

Where  $q_{Di,j}$  is the quantity of metal  $i$  downgraded to produce an alloy  $j$ , and  $\omega_{i,j}$  are weights used to penalized use of high quality metal in low alloy grade. In our case,  $q_{Di,j}$  and  $\omega_{i,j}$  will be:

$$q_D = \begin{bmatrix} 0 & q_{DA,B} & q_{DA,C} \\ 0 & 0 & q_{DB,C} \\ 0 & 0 & 0 \end{bmatrix} \quad \omega = \begin{bmatrix} 0 & 3 & 6 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{bmatrix}$$

The genetic algorithm will be in charge of generating sequences and the simulation model will evaluate the cost function for each sequence.

Table C.1 – Quantity and product details

Job	Quantity (T), product	Job	Quantity (T), product
1	16, A	11	14, B
2	24, B	12	16, B
3	16, A	13	24, C
4	14, A	14	20, C
5	24, A	15	16, C
6	20, A	16	18, C
7	8, A	17	22, C
8	23, B	18	17, A
9	20, B	19	15, B
10	17, B	20	11, A

#### C.4.1.1 Simulation models

Figure C.4 gives a high-level view of the simulation model structure used to replicate the aluminum manufacturing process. The one modeled contains 240 pots, 20 crucibles, two mixing furnaces and one waiting furnace.

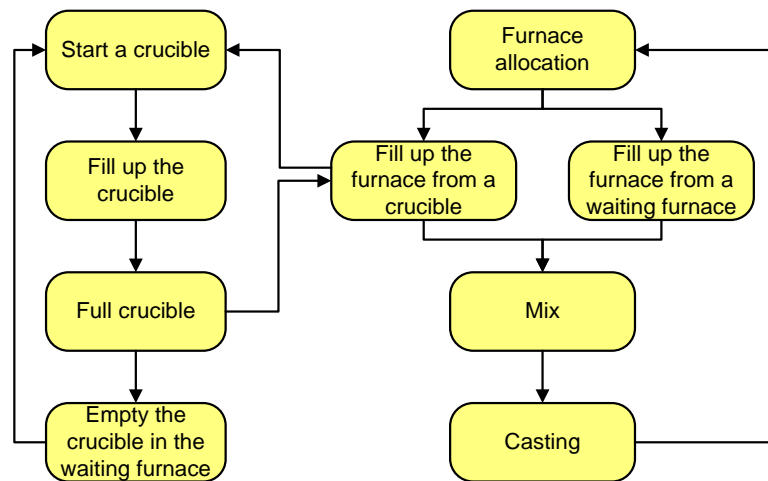


Figure C.4 – Simulation model

The pot lifecycle is synchronized by the tap time. When this happens, the pot is tapped in an empty crucible or in a crucible filled with aluminum of the same quality. This decision depends on the chemical properties of the pot, the actual time, and the filling level of the crucible. When a crucible starts to be filled with aluminum, it can only be used during a specific time range. During this time range, the crucible can be used by a mixing furnace to complete a production order. When the time range is over, the crucible has to be emptied into the waiting furnace. The aluminum quality contained in the waiting furnace always corresponds to the lowest aluminum quality emptied in the furnace (ie. if you emptied an high quality aluminum in the waiting furnace containing bad quality aluminum, all this aluminum is considered to be bad quality aluminum). There is no time restriction to use the waiting furnace aluminum.

When one of the two mixing furnaces is empty, it will be allocated to the first production order of the remaining sequence. The alloy quality to be produced will constrain the aluminum quality that can be used to complete this order. Only aluminum with the corresponding (or higher) quality can be used. When higher quality aluminum is used, it is a downgrade use and the plant tries to minimize this use. The mixing furnace can be filled with aluminum from crucibles or from the waiting furnace. This selection depends on the chemical properties of the available

aluminum and minimizes a penalty function (Table C.2). When the required quantity is in the furnace, aluminum is mixed with other components during a certain period of time, depending of the quantity of alloy to be produced. Then the alloy is cast.

Table C.2 – Penalty function example with three qualities of aluminum and alloy

Aluminum available → ↓ Alloy to be produced	A	B	C
I	0	-1	-1
II	5	0	-1
III	10	5	0

In the model, two kinds of downgrades exist. Figure C.5(a) illustrates a direct downgrade case. One unit of A is downgraded in B and C and one unit of B is downgraded in C. The second step of Figure C.5(b) shows an indirect downgrade. The unit of A downgraded in B is then downgraded in C. This case exists when a crucible of aluminum A is emptied in the waiting furnace containing aluminum B and then is used for an order requiring aluminum C. The model is designed to track the two kinds of downgrades.

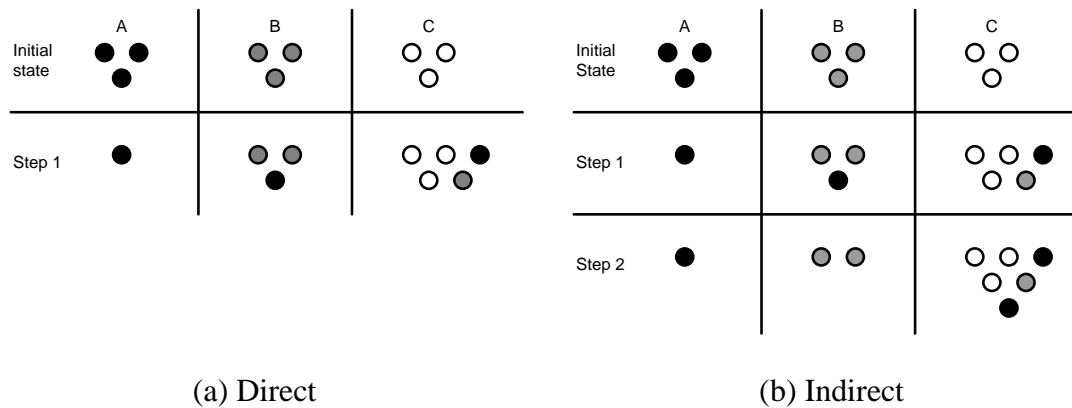


Figure C.5 – Downgrade

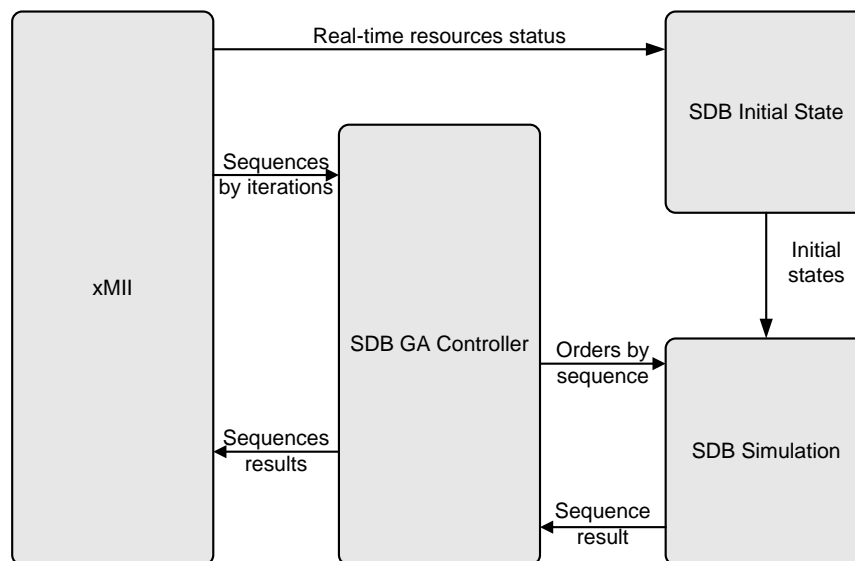


Figure C.6 – Simulation models and interactions

The simulation model has also to interact with the genetic algorithm. This interaction has two impacts on the model. First, the model has to be able to run the different instance representing the genetic algorithm population and send back results concerning each member of this population. Then, the model has to be initialized with the real-time resource statuses. To simplify the model development, we choose to separate the logic due to the genetic algorithm interaction, the real-time initialization synchronized with the shop floor statuses and the aluminum manufacturing process. In fact, we don't have just one simulation model, but three communicating together using the SD Builder message functionality (Figure C.6). The first model, called the controller model, is the only one to interact with the genetic algorithm and act as a controller for the plant model. The second one is responsible for keeping the resource statuses up-to-date in real time and communicating only with the MES and the two other models. Finally, the third model simulates the plant logic. When the optimization process starts, the controller model sends a message to the plant model to initialize the resources as they currently are on the shop floor. Then, the controller receives all the sequences for one iteration. Each sequence is sent one after the other to the plant model to be tested and the plant model returns the cast function value for this sequence. Then the

controller sends back the results to the genetic algorithm and waits for the next iteration. The message functionality is also used to directly update the model configuration when a major shop floor status change occurs (ie. a furnace breakdown). The model will take it into account immediately and send the genetic algorithm a message back to specify an initial conditions change. This last message indicates to the genetic algorithm to re-initialize the search process.

#### **C.4.1.2 Genetic algorithm**

In terms of an optimization problem, the genetic algorithm approach is summarized as follows. At any given point in time, the genetic algorithm generates a population of possible candidate solutions. Initially, the population size is chosen randomly. However, this choice typically depends on the characteristics of the problem. Each population component is a string entity of chromosome which represents a possible solution to the problem. The population components are evaluated based on a given fitness function. Highly fit population components are given the chance to reproduce through a crossover process with other highly fit population elements by exchanging pieces of their genetic information. This process produces “offspring,” or new solutions to the optimization problem based upon the high-performance characteristics of the parents. Premature loss of important information by randomly altering bits within a chromosome is prevented by a mutation process. This procedure continues until a satisfactory solution is achieved.

To solve the problem under study, we have developed a GA program based on an existing Toolbox (Chipperfield *et al.*, 1994). The main data structures in the GA toolbox are chromosomes, phenotypes, objective function values and fitness values. The chromosome structure stores an entire population in a single matrix of size  $N_{ind} \times L_{ind}$ , where  $N_{ind}$  is the number of individuals and  $L_{ind}$  is the length of the chromosome structure. Phenotypes are stored in a matrix of dimension  $N_{ind} \times N_{var}$  where  $N_{var}$  is the number of decision variables. A  $N_{ind} \times N_{obj}$  matrix stores the objective function values, where  $N_{obj}$  is the number of objectives. Finally, the fitness values are stored in a vector of length  $N_{ind}$ . In all of these data structures, each row corresponds to a particular individual.

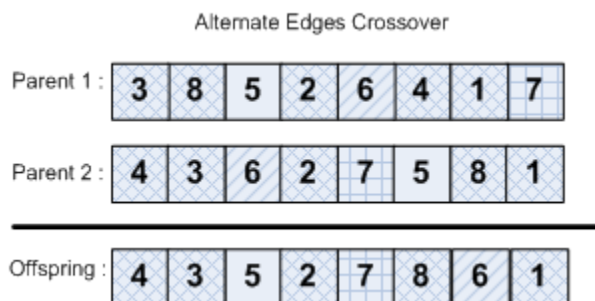


The GA toolbox uses MATLAB matrix functions to build a set of versatile routines for implementing a wide range of genetic algorithm methods. The following steps summarize the employed Genetic Algorithm:

- 1- Population representation and initialisation: permutation coding representation.



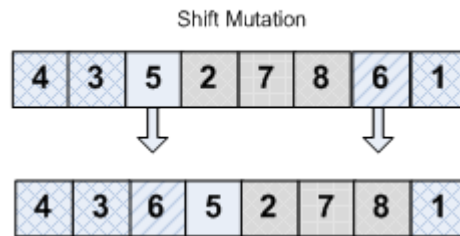
- 2- Fitness: the linear-**ranking** method of Baker (Baker, 1985).
- 3- Selection: roulette wheel selection (Goldberg, 1989).
- 4- Crossover: alternate edges crossover (Poon and Carter, 1995) with crossover probability « **Pc** », a crossover method for ordered chromosomes. It recombines pairs of individuals with given probability to produce offspring with respect to the existing jobs.



Here, a starting edge  $(i,j)$  is selected at random in one parent. Then the tour is extended by selecting the edge  $(j,k)$  in the other parent. The tour is progressively extended in this way by alternatively selecting edges from the two parents. When an edge introduces a cycle, the new edge is selected at random (and is not inherited from the parents).

- 5- Mutation: shift mutation (Chang *et al.*, 2007) with probability  $P_m$ . A mutation method for ordered chromosomes.

In shift mutation, two positions are randomly selected at first. As shown by the following figure, positions “5” and “6” are selected. The next step is to insert “6” right before “5” and “2”, “7”, “8” shift backward.



Let “**MaxGen**” be the maximum number of generations if the stopping algorithm rule is fixed following this criteria.

### C.4.2 Systems workflow

Figure C.6 presents the process workflow of the decision model within the proposed execution system architecture. In this example, the execution system is composed of four main components: the ERP system (SAP R/3), the MES system, the integration component (xMII), and the simulation/decision support system (figured by DSS).

At the beginning of the production cycle, the master plan is generated and recorded in the ERP system before being sent by xMII to the MES. xMII use Java connector (JCo) and BAPI to export data from R/3 and XML files or direct insert in the database to export data to the MES. When a new order arrives, the order sequence is updated. As the casting of a batch ends, the produced quantity is measured to identify any production shortfalls and update the production order schedule. When those events are captured, the decision making process is called. All data required to initialize the genetic algorithm and the simulation model are provided by xMII as CSV files. At each decision making time point, xMII is used to update the schedule and dispatch it. The allotted time period for decision making is fairly short (within a minute).

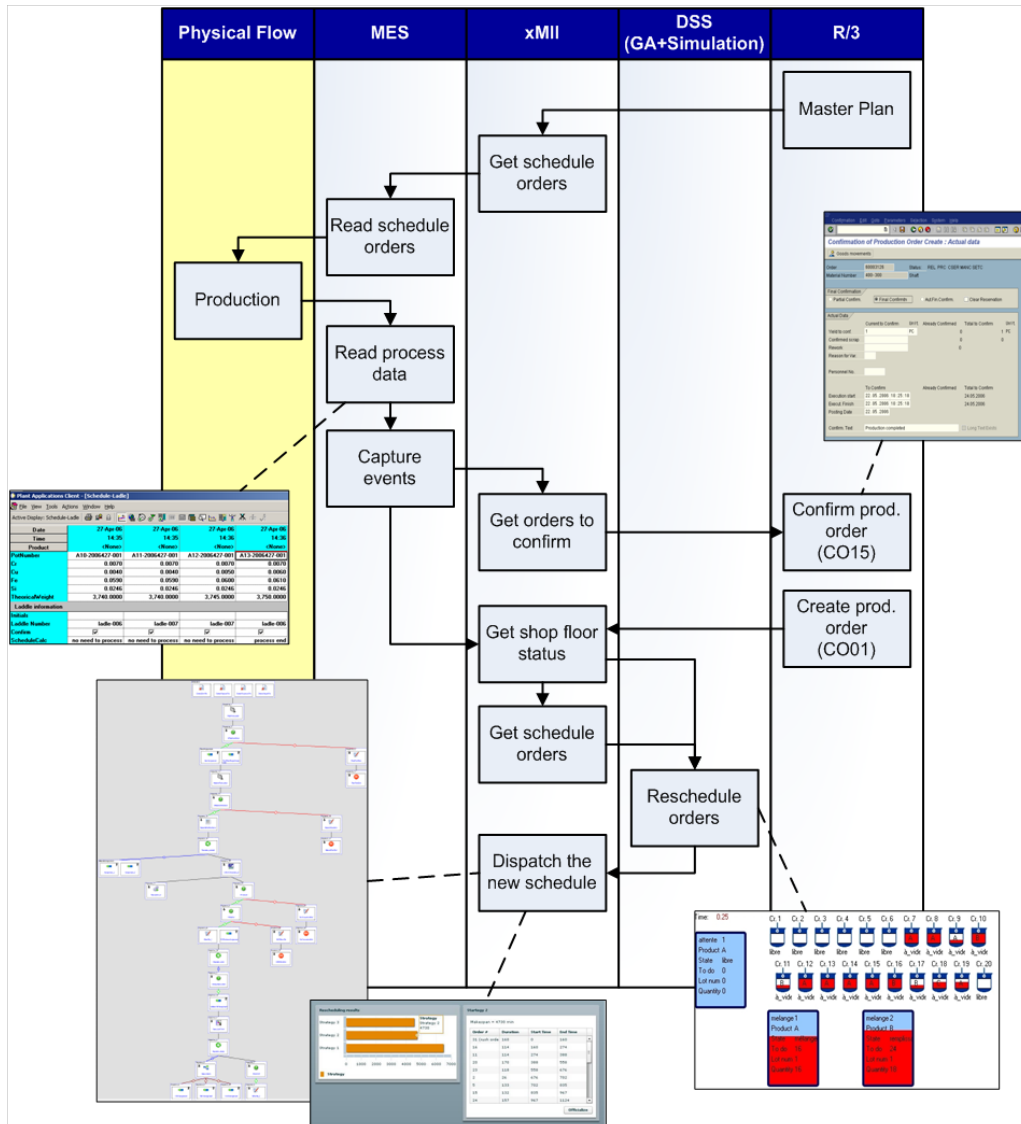


Figure C.7 – Scenario system workflow

### C.4.3 Validation

It is extremely difficult to compare the performance of different evolutionary algorithms since most researchers use their own instances of test problems, i.e., problems in which the processing times and due dates of the jobs are selected randomly out of a uniform distribution. To validate our model, we use a standard benchmark problem taken from Taillard (1993). This flow shop instance has 20 jobs and 5 machines, and uses the time seed 379008056. A calculated lower bound is 1290 and the best make span found by Taillard is 1359. With our model we found 1360 after 100 iterations and with  $P_c$  equal to 0.6. Being very close to Taillard results, we decided to go further with the developed genetic algorithm and the problem under study. It is interesting to note that when dealing with genetic algorithms, the choice of the GA parameters (i.e.,  $P_c$ ,  $P_m$ ) is an important issue to be taken into account since it can affect the optimization process and the final results. In the research literature, the choice of these parameters is generally based on experience.

### C.4.4 Results – discussion

Currently, orders are sequences based on a priority given by the planner. In Table C.1, orders are already classified by priority and it will be the reference sequence of our tests. The results for this sequence are presented in Table C.3. The total make span of the solution is given for informative purposes.

Table C.3 – Reference sequence results

Cost function	$q_{DA,B}$	$q_{DA,C}$	$q_{DB,C}$	Make span
583.6	46.8	51.6	66.8	418.75

The results obtained, including the optimization process, are given in Table C.4. Those results were obtained with  $N_{ind} = 20$  and  $MaxGen = 40$ . The weighted downgrade was reduced up to 46 % while the make span decreased by 12 %. As the following section will show, the Genetic Algorithm is fast to converge since the model is very constrained.

Table C.4 – Reference sequence results (GA)

Cost function	$q_{DA,B}$	$q_{DA,C}$	$q_{DB,C}$	Make span
315.4	18.2	20.4	69.2	367.75

The proposed cost function evaluated by the simulation model is centered on the plant's needs to minimize the downgrade. But it is well known that a planner never bases his choice on a single objective, and he also has other constraints such as customer due dates. If we want, the genetic algorithm proposes a new schedule under the downgrade and due date constraints, and we should update the cost function to integrate the tardiness. Equation (1) combines the actual weighted downgrade with the weighted quadratic tardiness. The weights can be used to model the fact that some jobs are made to order and others are made to stock. The smelter schedule is synchronized with the delivery planning and with the cost function (1) we will be able to propose schedules more respectful to the global problem.

$$F' = \alpha \times \left( \sum_i \omega_i^{DD} T_i^2 \right) + (1 - \alpha) \times \left( \sum_i \sum_j \omega_{i,j} \times q_{Di,j} \right) \quad (3)$$

with:  $T_i = \max(0, C_i - D_i)$

$C_i$  : job  $i$  completion time

$D_i$  : job  $i$  due date

For  $i$  in  $J$ , the set of jobs to schedule.

### C.4.5 Optimization process performance

As we propose an executive architecture, it is important to measure the performance of the optimization process integrated in the ERP-controlled environment. This performance can be expressed as the time used by the process to return a solution and can be decomposed as an ERP

time, a simulation time and a CPU time. The ERP time is the time spent to retrieve the required data to initialize the simulation model and the GA, to send the data to and receive the results from the optimization nucleus and to update the orders' start date. This time can fluctuate from 30 to 45 seconds depending on the network and the ERP server load. The simulation time is the time used by the optimization nucleus to load the sequence sent by the GA, simulate each sequence and send the results to the GA. The CPU time is the time used by the GA to generate new sequences. The CPU and simulation times are related and depend on the number of sequences generated at each step of the GA (i.e.,  $N_{ind}$ ) and the number of steps the GA processes (i.e., MaxGen) to provide a solution. Figure C.8 provides a computation time analysis of the problem using jobs presented in the Table C.1.

Table C.5 – Comparative study

Cases	MaxGen	Nind	Execution Time	Cost function	Execution Time Variation	Cost function Variation
Basic	10	20	763	368.4	---	---
2	20	20	1409	342.4	↑ 84.67 %	↓ 7.06 %
3	30	20	2044	315.4	↑ 167.89 %	↓ 14.39 %
4	40	20	2742	315.4	↑ 259.37 %	↓ 14.39 %

As shown by Table C.1, the GA parameters  $N_{ind}$  and MaxGen have a major impact on the time spent to solve the problem. It is therefore important to find the proper sets for each plant depending on the size of the problem and the planner's needs. At the same time, the election of the GA parameters influences the final solution (i.e., minimize downgrade). Hence, the same sensitivity analysis was conducted to study this relationship. As shown by Table C.5, the best values of the cost function are reached and stabilized around  $N_{ind}=20$  and  $MaxGen=30$ . Consequently, planners should make a choice between a weighted downgrade equal to 368.4 (reached after 12 minutes), and the best weighted downgrade equal to 315.4 (reached after 30 minutes) for a problem composed of 20 orders and 283 machines.

The simulation nucleus using three simulation models in place of a complex one is the business logic separation: one for the genetic algorithm, one for the real time synchronization, one for the plant simulation. The main advantage is the ease of designing and maintaining the models. The controller model is also completely independent of the business logic and can be reused for other purposes. A drawback of this system is the high quantity of messages generated. For each iteration, the genetic algorithm sends  $N_{ind}$  sequences of  $n$  jobs. Then for each sequence, the controller model sends the sequence to the plant models and forwards the results to the genetic algorithm. If we don't consider messages required to initialize the plant model, depending on the complexity of the shop floor modeled, there are at least  $2 \times (N_{ind} + 1) \times MaxGen$  messages sent and received by the simulation models. 100 iterations of 100 sequences of 20 orders represent 20,000 messages. Figure C.7 represents the messages' processing time. It is not exactly the time required to process the messages during the optimization process, because considering this non-negligible amount of time, the simulation models are designed to process tasks in parallel. For example, during an iteration, when the controller model receives a sequence and if the plant model is not already occupied to simulate a sequence, the sequence is immediately sent to the plant model.

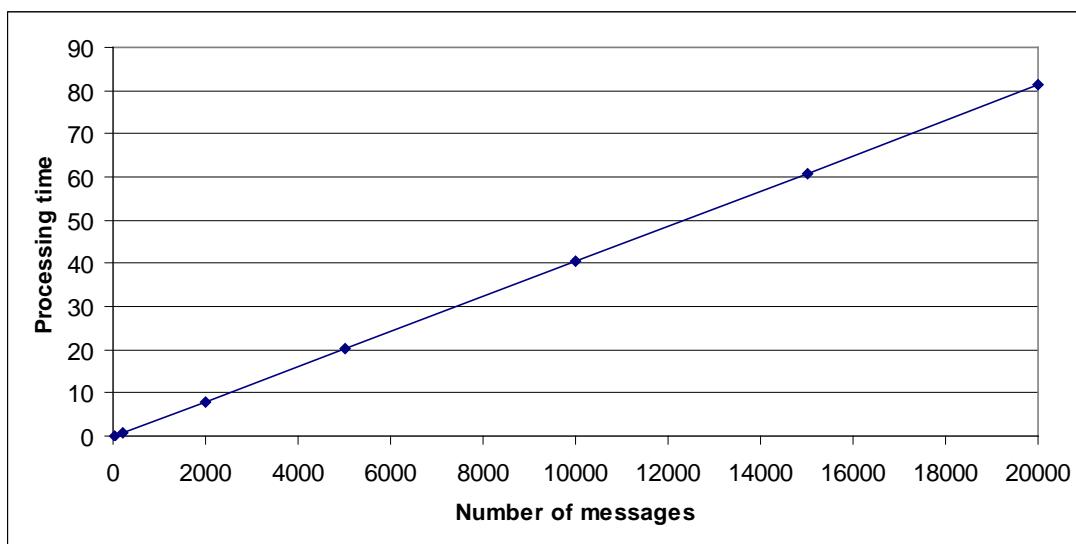


Figure C.8 – Messages processing time

## C.5 Conclusion

In this paper, we presented a generic manufacturing execution platform supporting real-time decision making. The proposed execution platform complements the traditional ERP/MES model to simulation-based real-time optimization models. To illustrate the functionalities and the interactions between the different components of the execution platform, we presented a demonstrative example based on a real scenario in the aluminum industry. This example illustrates how a genetic algorithm and a real-time discrete event simulation model can be integrated within common information systems used by manufacturing organizations.

Even if the demonstrative example has revealed the usefulness and robustness of the proposed execution platform, we claim that other scenarios should be studied in the future to demonstrate the generic aspect of our approach in solving real-time decision-making problems of adaptive manufacturing systems. Future works will also involve the integration of other optimization techniques to improve the simulation-based real-time optimization unit. Moreover, to deal with the complexity behind an efficient integration of the components involved, other approaches are currently being investigated.

## C.6 References

- Azadivar, F., and Lee, Y.-H. (1998). Optimization of discrete variable stochastic systems by computer simulation. *Mathematics and Computers in Simulation*, 30(4), 331-345.
- Baker, J. E. (1985). *Adaptive selection methods for genetic algorithms*. Paper presented at the Proceedings of the 1st International Conference on Genetic Algorithms.
- Carson, Y., and Maria, A. (1997, Dec 7-10 1997). *Simulation optimization: Methods and applications*. Paper presented at the Proceedings of the 1997 Winter Simulation Conference, Atlanta, GA.



- Chang, P.-C., Hsieh, J.-C., and Wang, C.-Y. (2007). Adaptive multi-objective genetic algorithms for scheduling of drilling operation in printed circuit board industry. *Applied Soft Computing Journal*, 7(3), 800-806.
- Chaudhry, S. S., and Luo, W. (2005). Application of genetic algorithms in production and operations management: A review. *International Journal of Production Research*, 43(19), 4083-4101.
- Chipperfield, A. J., Fleming, P. J., and Fonseca, C. M. (1994, 21-22 September). *Genetic algorithm tools for control systems engineering*. Paper presented at the Proceedings of Adaptive Computing in Engineering Design and Control, Plymouth Engineering Design Center (UK).
- Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Boston, MA, USA: Addison-Wesley.
- Iassinovski, S., Artiba, A., and Fagnart, C. (2008). A generic production rules-based system for on-line simulation, decision making and discrete process control. *International Journal of Production Economics*, 112(1), 62-76.
- Kouiss, K., and Pierreval, H. (1999). *Implementing an on-line simulation in a flexible manufacturing system*. Paper presented at the Simulation in Industry'99. 11th European Simulation Symposium 1999. ESS'99.
- Poon, P. W., and Carter, J. N. (1995). Genetic algorithm crossover operators for ordering applications. *Computers & Operations Research*, 22(1), 135-147.
- Ranga, P., and Dwivedi, S. N. (2003). *Research issues in responsive and agile manufacturing*. Paper presented at the International Conference on Agile Manufacturing, Advances in Agile Manufacturing, ICAM 2003, Beijing, China.
- Ruiz, R., Carlos Garcia-Diaz, J., and Maroto, C. (2007). Considering scheduling and preventive maintenance in the flowshop sequencing problem. *Computers and Operations Research*, 34(11), 3314-3330.

Saad, S. M., and Gindy, N. N. (1998). Handling internal and external disturbances in responsive manufacturing environments. *Production Planning and Control*, 9(8), 760-770.

Shin, F., Ram, B., Gupta, A., Yu, X., and Menassa, R. (2004). *A decision tool for assembly line breakdown action*. Paper presented at the 2004 Winter Simulation Conference.

Taillard, E. (1993). Benchmarks for basic scheduling problems. *European Journal of Operational Research*, 64(2), 278-285.

Wadhwa, S., Saxena, A., and Chan, F. T. S. (2008). Framework for flexibility in dynamic supply chain management. *International Journal of Production Research*, 46(6), 1373-1404.

## **ANNEXE D: AN IMPROVED GENETIC ALGORITHM APPROACH FOR ON-LINE OPTIMIZATION PROBLEMS**

Référence : Saenz de Ugarte, B., Artiba, A., & Pellerin, R. (soumis en octobre 2009). An Improved Genetic Algorithm Approach for On-line Optimization Problems. *Production Planning & Control*.

**Abstract** – Real-time systems have to react to disruptive events in manufacturing environments within tight time constraints. This paper aims to enhance standard genetic algorithms to reduce the execution time and improve the solution quality when the objective function is expensive to evaluate. More precisely, we propose the addition of a cache memory to avoid re-testing an already tested solution and a predator mechanism to relax evaluation. After discussing the applicability conditions of those mechanisms, we propose a demonstrative example, based on a real scenario in the aluminium industry, in which a genetic algorithm and a real-time discrete event simulation model are integrated to support the rescheduling process in ERP-controlled environments.

**Keywords** – Genetic algorithm, rescheduling, real-time decision, predator, ERP.

## D.1 Introduction

Scheduling production activities involves the determination of work orders and of the corresponding tasks to be carried out on different equipment units, as well as the detailed timing of all tasks taking into consideration a number of objectives and constraints. The scientific literature provides a number of methods designed to optimize some performance criteria (plant profitability, make span, costs, etc.), assuming that all relevant parameters are already known and will not change during the scheduling horizon. In reality, shop floors are seldom stable for longer than half an hour. Constraints evolve and targets move. Information in enterprise systems is often not completely up to date, and schedule updates are often performed by human schedulers.

As computing power increases, real-time control is gaining much more attention and is generating considerable interest in the academic and industrial communities. Theoretically, real-time optimization implies instantaneous actions. However, absolute real-time actions are nearly impossible in practice. In addition, classical optimization approaches are unable to solve complex problems found in stochastic manufacturing systems. A realistic approach suggests the development of algorithms capable of proposing feasible solutions within a time frame short enough to support on-line use.

In that context, simulation modelling is a good alternative to take into account the dynamic and stochastic behaviour of a manufacturing system. In the literature, simulation based optimization approaches can be classified in different categories (gradient based search method, stochastic optimization, statistical methods, etc.) but the most encountered category used in real-time decision making is the heuristic methods (for more details, readers should consult (Carson & Maria, 1997)). Heuristic methods consist of a random exploration of the admissible solutions in the whole decision's space. The search process ends when the best solution is found. At each point in the search process, the objective function value of the problem is estimated via the simulation model. Thus, no information regarding the analytic form of the objective function is required. Some common approaches cover simplex search (Azadivar & Lee, 1988), tabu search, simulated annealing (Lee & Iwata, 1991; Ogbu & Smith, 1990), genetic algorithms (GA), and evolutionary strategies (Chaudhry & Luo, 2005).

It is of interest to note that previous research and surveys (Azadivar & Tompkins, 1999; Chaudhry & Luo, 2005; Ruiz et al., 2007) have demonstrated the effectiveness of genetic algorithms and evolutionary strategy solutions. Genetic algorithms (Goldberg, 1989; Holland, 1975) are categorized as global search heuristics and are a particular class of evolutionary algorithms that use techniques inspired by evolutionary biology. Candidate solutions to the problem are represented as chromosomes. The genetic algorithm then creates an initial population of solutions, uses an evaluation procedure to know the performance of each chromosome, and applies artificial genetic operators such as mutation and crossover to evolve the solutions in order to find the best one(s).

Up until now, scientific literature on this subject has been classified into three categories. The first category demonstrates the successful use of GA to solve various problems in various fields such as scheduling (Zobolas et al., 2008), maintenance (Mohagheghi & Shayegan, 2009) or civil engineering (Farhat et al., 2009). Even when genetic algorithms are applied to real world problems, the resolution time is never a major concern; this is surely because the contribution of the papers is more the problem to solve than the GA itself. GA is used because it is a simple and efficient algorithm with good convergence properties even when its parameters are not finely

tuned. However, the realistic nature of these problems should imply a lot of variables and increase the resolution time. The second category proposes new crossover (Tsai et al., 2008; Yaurima et al., 2008) and mutation (Deep & Thakur, 2007) operators in order to avoid inconsistency while generating a new chromosome. Depending on the problems to be solved and constraints to deal with, chromosomes and genes can be coded in different ways, and some operators can be applied and others cannot. In the research literature, authors have proposed new and more effective operators to deal with those specific conditions. They have used various performance criteria to judge the efficiency, accuracy and reliability of those operators applied to benchmark test problems, but the resolution time is not the concern of those studies. Those operators will not decrease the resolution even if they can strongly influence the convergence property. The focus of these operators is the GA itself and not the chromosome evaluation process. Finally, the last category includes all research aiming at enhancing the effectiveness of the GA method by proposing new approaches such as parallelization (Defersha & Chen, 2008; Isgro & Tegolo, 2008), coevolution (Gu et al., 2008; Potter & De Jong, 1994), hybridization (Cheng et al., 2009; Rossi & Boschi, 2009; Tavakkoli-Moghaddam et al., 2009; Whittaker et al., 2009; Zobolas et al., 2009). Here, GA is enhanced using more than one population of chromosomes or combining the GA evolution process with other heuristics. Generally, these methods result in an increase in the number of chromosomes, and thus the number of evaluations. If there is a long evaluation time, the resolution time will not decrease.

In most cases, the optimization is processed off-line or the evaluation procedure is not the bottleneck in terms of time consumption. However, to solve large scale problems, it is sometimes necessary to enhance the efficiency of the algorithm (Sastry et al., 2004). Some common approaches include time continuation (Sastry & Goldberg, 2007) and evaluation relaxation (Sastry et al., 2006). These approaches are concerned with an efficient use of the time available to run the GA. Time continuation uses smaller populations running through several continuation epochs. This allows stopping, working with the current population and exploring fresh regions of the search space. Here, the number of evaluations is constant compared to a larger population running for a single or small number of epochs. Evaluation relaxation uses a

surrogate function that requires less computational time without affecting the run quality. The surrogate function is particularly used at the beginning of the run, when extremely accurate fitness values are not required, and can reduce the resolution time by up to 20%.

In this paper, we address the case in which the evaluation procedure of GA is time consuming. Under a hypothesis linked to objective function, we propose two mechanisms to reduce time consumption of the evaluation procedure, enhancing the convergence rapidity, and that can be easily integrated with existing GA based heuristics. The remainder of this paper is structured as follows. Section 2 describes the cache memory and predator mechanisms and the applicability conditions. In Section 3, the two mechanisms are applied to benchmark problems for validation. Section 4 presents a demonstrative example, based on a real scenario in the aluminium industry, in which a genetic algorithm and a real-time discrete event simulation model are integrated to support the rescheduling process in ERP-controlled environments. Finally, a brief discussion of future perspectives will conclude this paper.

## **D.2 GA enhancements**

The proposed enhancements assume that the evaluation process used by the GA is time consuming compared to other mechanisms integrated in the heuristic. The aim of those enhancements is to provide mechanisms to reduce this time consumption without degrading the effectiveness of convergence.

### **D.2.1 Cache memory**

In its more simple form, GA will randomly generate the initial population. This population will then evolve using crossover and mutation operators. In most cases found in the scientific literature, the chromosome evaluation process is never detailed and if the new chromosome had already been tested in a past iteration, the chromosome will be re-evaluated. This phenomenon occurs more frequently on the end of the evolution process. But in reactive real-time systems, computation time is a crucial constraint. This leads us to propose a cache memory mechanism.

This mechanism can be used in two different ways. The first way is to implement the memory just before the evaluation process (Figure D.1). When the evaluation of a chromosome is required, it is searched in the memory. If the chromosome is already in memory, the evaluation is directly returned to the GA. If not, the chromosome is evaluated and the result is put in memory and then returned. This approach has the main advantage of being implemented just at the interface between the GA and the evaluation process, so nothing needs to be changed in the two blocks.

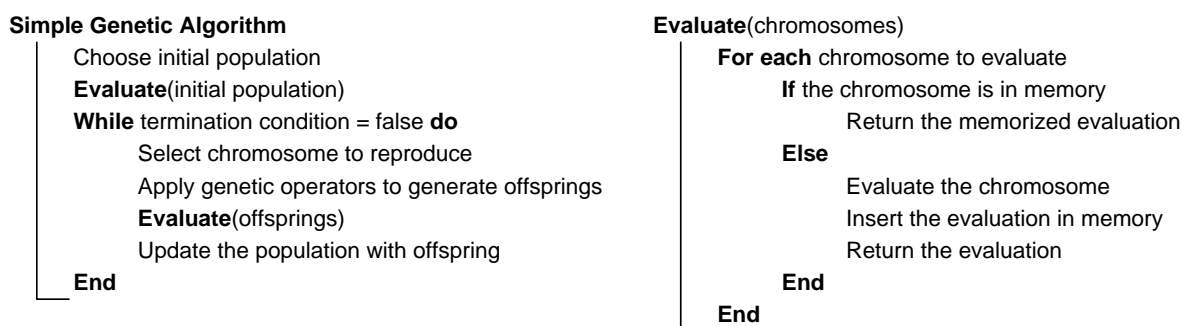


Figure D.1 – Simple Genetic Algorithm with memory

Using the memory is valuable if the time saved by the use of the memory is less important than the time spent to search the chromosomes in memory. This is generally the case when the evaluation process takes a lot of time. But when the GA runs in real-time and computation time is a constraint, we only have a finite number of evaluations and the GA may generate a chromosome already in memory. This chromosome may take the place of a really new chromosome, potentially a best one. The second way to use the memory is to force the chromosome to be unique in the GA instance history (Figure D.2).

This way of using the cache memory requires modifying the heuristic implementation but provides a more efficient search space exploration. It also ensures that each offspring is really new compared to its ancestors. With this memory implementation, we will not gain time on



evaluation because each chromosome is unique and will be evaluated. Fortunately, the number of iterations required may be reduced in order to obtain the same kind of results.

```

Genetic Algorithm with inter-generational memory
  Choose initial population (all initial chromosome unique)
  Evaluate(initial population)
  While termination condition = false do
    Select chromosome to reproduce
    Apply genetic operators to generate offsprings
    While offsprings are not unique do
      Apply genetic operators to generate offspring
    End
    Evaluate(offsprings)
    Update the population with offspring
  End

```

Figure D.2 – Genetic Algorithm with inter-generational memory

### D.2.2 Predator

The predator is an evaluation relaxing mechanism and the main part of the heuristic does not need to be modified. However, the objective function needs to verify two hypotheses:

1. The objective function has to be evaluated step by step.
2. The objective function is a non-decreasing function over steps when minimizing the objective function.

These hypotheses are not too strong and are verified, for example, when simulation is used in the evaluation process and when the objective is to minimize the total make span.

For this mechanism, the chromosome acts as a prey and a predator is hidden in the evaluation process (Figure D.3). The predator is a combination of three variables: a predation threshold  $T_{pred}$ , a predation probability  $P_{pred}$ , and a default value to return  $M_{pred}$ . The two first variables can be static or dynamic. For example, the best-known value of the objective function can be used as the

predation threshold. During the chromosome evaluation, if the objective value becomes greater or equal to  $T_{pred}$ , the predator will act with the probability  $P_{pred}$ , abort the evaluation and return  $M_{pred}$ . The predator has only one chance per evaluation to kill the chromosome.

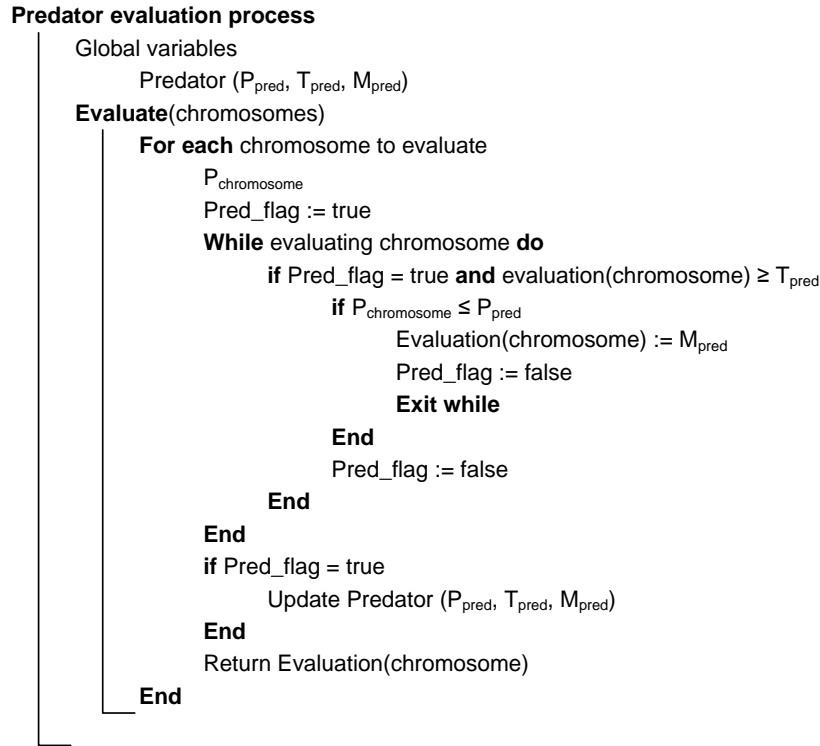


Figure D.3 – Predator evaluation process

With the predator mechanism, the evaluation of a poor quality chromosome will be aborted. Practically,  $P_{pred}$  and  $T_{pred}$  have to be fine-tuned to avoid premature convergence to a local optimum and to keep some diversity in the population (at least for the initial one). For example, if you choose the best-known value of the objective function as  $T_{pred}$ , if  $P_{pred}$  is high and if a good chromosome is evaluated at the beginning of the evolution process, there is a high probability that the predator will kill all of the chromosomes evaluated after this one. The demonstrative example will provide an insight about threshold functions.

### D.3 Validation

Before implementing the cache memory and the predator in a study based on a real life scenario, the two mechanisms were validated on benchmark problems. This validation aims to quantify how much these enhancements are valuable and also to highlight GA parameters that can positively and negatively influence their performance. To do that, we use the jobshop generator proposed by Taillard (1993). The problems used are defined by Table D.1 parameters. A generated jobshop problem is completely defined by the number of jobs  $N_{jobs}$ , the number of machines  $N_{machines}$  and the machine and time seeds used by the random generator. The chromosome will represent the job priority: chromosomes have a size of  $N_{jobs} \cdot N_{machines}$ .  $N_{machines}$  is used to influence the time required to compute the makespan of the chromosome.  $N_{machines}$  are chosen to fast (less than 0.1 second), medium (around 5s) and long (around 10s) evaluation times. For each problem size, we generated 5 different problems. Population size, number of iterations and generation gap influence the number of chromosomes tested for each GA run. The mutation probability has an influence on the stochastic exploration of the search space. Each problem will be tested without memory and without a predator, with memory only, with predator only and with memory and a predator. For this study, the predator uses the best-known value for the objective function plus a static tolerance of 5%. We use two predators with different predation probabilities. Finally, five replications are run for each set of parameters for a total of 72900 GA runs.

In its primal form, a memory is used to avoid retesting an already tested solution. This use of memory will be valuable only if the evaluation time of a solution is greater than the time required to search for the solution in memory, but under the long evaluation time hypothesis, this assumption is always true. The uniqueness chromosome rate determines how much time can be saved. The lower this indicator is, the more valuable the memory is. This indicator globally decreases when the size of the chromosome increases and when the number of tested sequences increases. With a small chromosome and at least 100 evaluations, it is easy to save 10% and more of the total runtime, with a bigger one, it can be 5% and more.

Table D.1 – Problem parameters

Parameter	Set of value
Number of jobs $\times$ Number of machines	{(20, 20), (20, 6500), (20, 12000), (50, 20), (50, 1700), (50, 3000), (150, 20), (150, 140), (150, 340)}
Seed time $\times$ Seed machine	{(302034063, 1203569070), (1437643198, 1692025209), (1792475497, 1039908559), (1647273132, 1012841433), (696480901, 1689682358)}
Population size	{20, 40, 100}
Iterations	{20, 50, 80}
Generation gap	{0.1, 0.3, 0.5}
Mutation probability	{0.3, 0.7}
Memory	{without memory, with memory}
Predator	{without predator, with predator}
$P_{pred}$	{0.4, 0.7}
GA seed	{957638, 162587311, 965299017, 687896268, 213027331}

When the memory is used to ensure the uniqueness of generated solutions over a generation, the runtime stays the same as the number of tests is constant. There is a small increase in the total runtime, resulting in the use of the memory, but it is less than 3% of the total runtime. The user will have to choose between the time required to search for a solution in memory on the one hand, and a better exploration of the search space on the other hand. For a given number of evaluations, the use of this kind of memory enhances the best-found solution in 55% of cases. This success rate increases with the number of tested sequences and when the uniqueness rate is low. This rate increases to 75% of cases when the GA does not find a good solution at the beginning of the run. This rate also increases when the number of iterations increases.

The predator mechanism was designed to reduce the run time without impacting the convergence property of the GA. Depending on the predator implementation, there is a risk with bigger

chromosomes to have a lower time gain due to an increasing number of  $T_{pred}$  tests. The main factor impacting the runtime is the predation probability  $P_{pred}$ . The runtime will decrease when  $P_{pred}$  increases because of a higher number of evaluation relaxations. When  $P_{pred}$  is equal to 0.4, the run time is reduced an average of 9.2%, and 17.7% when  $P_{pred}$  is equal to 0.7. In the vast majority of our tests, the best found solution stays the same as the problem with and without the predator, but in some cases, the best found solution is better. There is no correlation between the tested parameters to explain this fact, but as the predator relaxes the poor quality chromosome evaluation, the GA will focus on the better ones, and in some cases, it can result in a solution enhancement.

The combination of the inter-generational memory and predator mechanisms aims to reduce the run time while enhancing the best found solution. The idea is to compensate for the time lost during the memory exploration with the time gained by the predator. Parameters influencing the results are the same. The percentage of runs with a better found solution is the same for the memory only, but the gain on the run time increases (-9.7% when  $P_{pred}$  is equal to 0.4 and -21.4% when  $P_{pred}$  is equal to 0.7). It is important to note that when the GA without memory and without a predator found a good solution early in the run, this solution was also found with the memory and the predator but the run time was drastically reduced (up to 40%).

In summary, the memory will be more efficient when the chromosome size is small or when the number of tested chromosomes is high (large population size or generation gap or number of iterations). The inter-generational memory seems to be more efficient when the GA does not find a good solution early during the run. There is more of a chance to be in this situation if the initial population size is not too large. Finally, the predator is more efficient with a small chromosome size, but can be easily adapted for a bigger size. A higher value of  $P_{pred}$  trends results in higher gains on the runtime.

Due to the wide scope of the problem size study here, the gain on these two mechanisms can be difficult to evaluate. To demonstrate the applicability of the proposed approach, in the following

section we present a case study based on a real world scheduling problem in the aluminum industry.

## **D.4 Case study**

The aluminum transformation process is a hybrid continuous and discrete manufacturing process. In the early stage of the process, anodes made of oil coke and tars are bathed in a hot electrolyte. The electrolyte then dissolves the alumina and the aluminum is collected in pots. The chemical reaction on anodes takes many days (continuous process). When a pot is full, it is then siphoned in a crucible (discrete process). The crucible is used to transfer the melting aluminum from pot lines to a furnace. In the furnace, alloys are prepared and mixed. At the end of the process, the alloy is transferred to the cast line to make alloy bars.

In our example, a pot line has a capacity of 240 pots. Pots are tapped individually every 24 hours. Each individual pot has chemical properties and a scheduling time. Chemical properties can change over time and are updated with lab values. Chemical properties include the level of metal contamination. Depending on contamination levels, some pots may not be used for producing high quality alloys (higher market value). When the metal is put into one of the 20 available crucibles, it is transported to the cast lines. Metal arrives in the cast house in an uncontrolled sequence, which depends on the real tapping schedule.

Batches of alloys are produced in the cast lines composed of two mixing furnaces. As metal always arrives in a non-controlled sequence, crucibles may need to be emptied into a waiting furnace. The chemical properties of the waiting furnace therefore vary according to the arrival and properties of new crucibles.

The operator of this manufacturing system aims at different objectives. At the corporate level, the objective is to optimize the allocation of alloys between plants according to the conditions and capacity of each plant. This problem is not the concern of this paper. On the plant level, the operator of this manufacturing system objective is to optimize the daily schedule according to the real-time shop floor conditions. The latter objective is measured by comparing the quality of the

metal produced to the quality sold to the customer. Globally, the objective of the shop floor personnel is to restrict the use of high quality metal for customer orders of a higher valued alloy and minimize the use of high quality metal for customer orders requesting low alloy grades.

Plant allocation of alloys and production orders are currently managed by the foundry ERP. The scheduler uses no particular tools to produce the detailed schedule. Orders are released in the ERP and the weekly schedule is sent to the shop floor. A MES system is used to track the production status and the aluminum quality at the earliest stage of the manufacturing process. When an order is partially or totally complete, the order status is updated in the ERP. When a new order is allocated to the plant, or when an order is partially completed (because of a lack of aluminum with a sufficiently high grade), the scheduler has to update the schedule that is made twice a week. In reality, decisions are made by the scheduler. Thus, the quality of the decisions depends strongly on his experience and the time available to make the schedule. This fact creates an important variance from one scheduler to another.

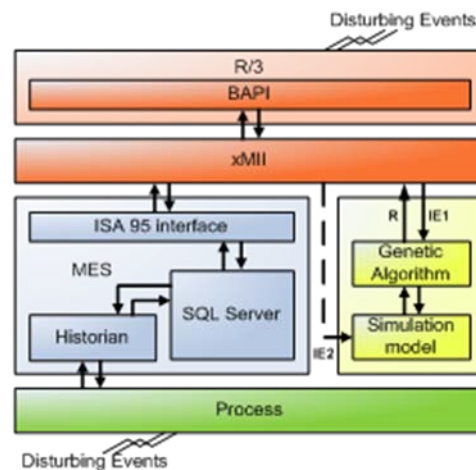


Figure D.4 – Real time execution architecture

The existing ERP-controlled environment is insufficient in guaranteeing a quick response to random events as they occur on the shop floor. Moreover, it falls short of meeting the quality objectives of the company. To bring a feasible solution that can be easily integrated into the

existing architecture, there is a need to complement it with real-time decision-making capabilities as shown on Figure D.4. The decision-making nucleus of our system is the combination of a genetic algorithm and a discrete event simulator. The simulator runs models of different parts from the controlled system. The genetic algorithm responds to incoming messages by initiating decision-making logic, depending on the disruptive event and the current system state.

In this scenario, mixing furnaces are the main constraints. While it is the bottleneck of the smelter, the planner only generates one production sequence for all of the furnaces. When a mixing furnace is free, it will process the first non-allocated order of the sequence. In the case study, 20 jobs are in the system and two of them are currently being processed by the mixing furnaces. The maximum load of a furnace is 24 tons, and there are three kinds of alloys ( $n = 3$ ).

The main objective of the optimization is to generate a sequence minimizing the use of high quality metal  $i$  for customer orders requesting low alloy grades  $j$ . The cost function used by the optimization model is as follows:

$$\min \sum_i \sum_j \omega_{i,j} \times q_{Di,j} \quad (1)$$

$$\omega_{i,j} = \begin{cases} (j-i)(n-i+1) & \text{if } j > i \\ 0 & \text{else} \end{cases} \quad (2)$$

Where  $q_{Di,j}$  is the quantity of metal  $i$  downgraded to produce an alloy  $j$ , and  $\omega_{i,j}$  are weights used to penalized use of high quality metal in low alloy grade. In our case,  $q_{Di,j}$  and  $\omega_{i,j}$  will be:

$$q_D = \begin{bmatrix} 0 & q_{DA,B} & q_{DA,C} \\ 0 & 0 & q_{DB,C} \\ 0 & 0 & 0 \end{bmatrix} \quad \omega = \begin{bmatrix} 0 & 3 & 6 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{bmatrix} \quad (3)$$



## D.4.1 Implementation

### D.4.1.1 Simple Genetic Algorithm implementation

To solve the problem under study, we have developed a GA program written in Ruby, a dynamic, reflective, general purpose object-oriented programming language. The main class objects are the GA itself and a chromosome. As the chromosome is encoded as a one-dimension array of size  $L_{ind}$ , the chromosome class inherits from the array class and we add class variables for the objective function value, the fitness value and the mutation probability. The GA class stores the population in a class variable as a one-dimension array of size  $N_{ind}$ , the number of individuals. At each iteration,  $N_{ind} \times G_{gap}$  offsprings are generated where  $G_{gap}$  is the generation gap.

The following steps summarize the employed Genetic Algorithm:

1. Population representation and initialization: permutation coding representation.

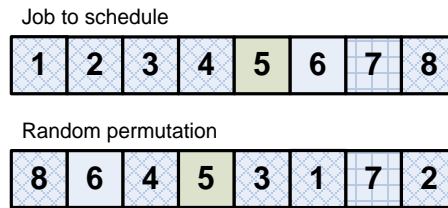


Figure D.5 – Permutation coding

2. Fitness: the linear-ranking method of Baker (Baker, 1985).
3. Selection: roulette wheel selection (Goldberg, 1989).
4. Crossover: two-point crossover is used, which sets two crossover points randomly. The section between the points is kept from one parent. For the outside sections, genes are the same of this parent but ordered as in the other parents (Gu et al., 2008).

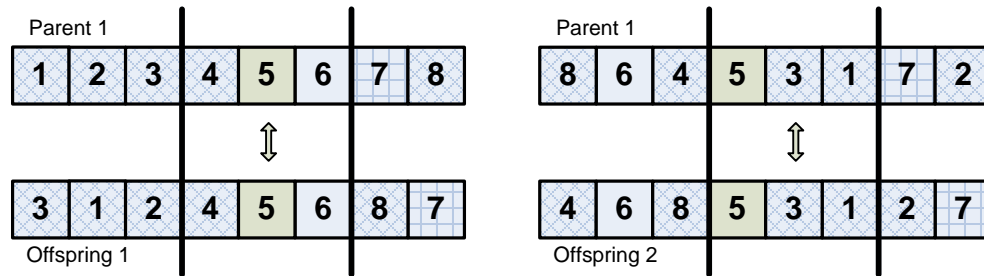


Figure D.6 – Crossover

5. Mutation: permutation of two randomly selected genes with probability  $P_m$ . This approach has avoided the issue of an illegal solution, and is a simple and effective method.

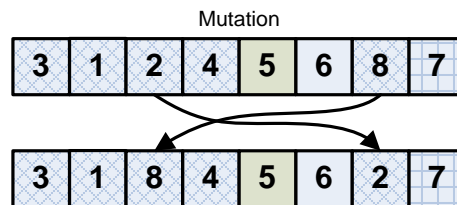


Figure D.7 – Mutation

Let  $MaxGen$  be the maximum number of generations if the stopping algorithm rule is fixed following this criteria.

#### D.4.1.2 Cache memory implementation

The cache memory is implemented in the GA class as a hash table of chromosomes. As chromosomes are arrays, an integer signature is calculated for each chromosome. Two identical chromosomes have the same signature if they are identical. Two chromosomes with a different signature are not equal but sometimes two different chromosomes can have the same signature. The signature is used as a key in the hash table and chromosomes with the same signature are stored in an array. To check if a chromosome is unique, we search the array corresponding to this

signature. If this array does not exist, the chromosome is unique. If this array exists, the chromosome is compared gene by gene with each chromosome of the array.

#### D.4.1.3 Predator implementation

In the demonstrative example, the predator is directly implemented in the simulation model. The predator uses a static predation probability  $P_{pred}$  with a dynamic predation threshold. The first implementation of the predator uses the best-known value for the objective function plus a static tolerance of 15%.  $T_{pred}$  is updated each time a better solution is found. This implementation is quite simple and has some shortfalls, but it will already provide a good impact on the evaluation time consumption, which we will present in the next section.

### D.4.2 Results and discussion

Currently, orders are sequenced based on a priority given by the planner. Orders are already indexed by priority. The results<sup>1</sup> obtained with a simple genetic algorithm (SGA) with and without a memory are given in Table D.2. Those results were obtained with  $N_{ind} = 20$ ,  $MaxGen = 40$ ,  $P_m = 0.6$  and  $G_{gap}=0.2$ . Those values will be kept constant during experimentations. The memory is used to ensure the singularity of each evaluated chromosome (Figure D.2).

Table D.2 – Simple Genetic Algorithm with and without cache memory

GA type	Best objective value fund	$\Delta_{objective}$	Runtime (s)	$\Delta_{runtime}$
SGA	291.0	-	709.917	-
SGA + memory	263.1	-9.59 %	688.711	-2.99 %

<sup>1</sup> Experimental computations were performed on a Sony Vaio VGN-N130QG with a 1.60GHz Intel dual core and 1Go of RAM on Windows XP SP3.

As expected, the memory used in this way allows a better exploration of the search space and for the same number of sequences generated. The cache memory allows a gain of 10 % on the best-found objective function value. The gain on runtime is not significant and due to the mean on all experimentations. It can be noted that in the case of the SGA, a mean of 12 % of evaluated sequences has already been evaluated in past iterations.

The SGA results are used as reference for the predator experimentation. Results obtained by the predator implementation described in Section 4.3 combined with the SGA with memory, for different values of  $P_{pred}$ , are given in Table D.3. As expected, as  $P_{pred}$  is increased, computation time decreases because more sequence evaluations are relaxed. There is also a significant gain of a mean of 19 % on the objective function, but there is no correlation between this gain and  $P_{pred}$ . If this parameter is kept static, experimentation will be needed to find the value of the most interest, depending on the evaluation process behind.

Table D.3 – Predator and Genetic Algorithm – Static tolerance

( $T_{pred}$  = best objective found  $\times$  115 %)

$P_{pred}$	Best objective value fund	$\Delta_{objective}$	Runtime (s)	$\Delta_{runtime}$
0.4	242.3	-16.72%	635.812	-10.44%
0.5	235.7	-19.01%	625.135	-11.94%
0.6	217.7	-25.18%	588.682	-17.08%
0.7	241.6	-16.98%	579.662	-18.35%
0.8	247.7	-14.87%	565.365	-20.36%
0.9	225.1	-22.63%	563.109	-20.68%
1	233.0	-19.93%	537.141	-24.34%

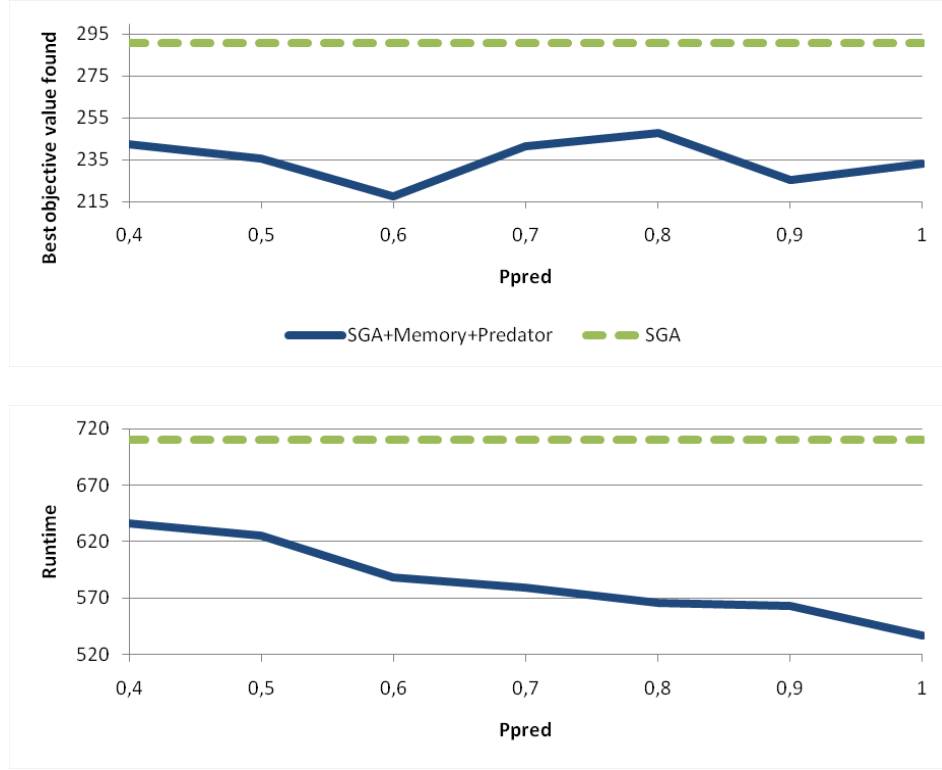


Figure D.8 – Predator and Simple Genetic Algorithm

$$(T_{pred} = \text{best objective found} \times 115 \%)$$

When analyzing the predator mechanism integrated with the SGA, we can have better results on the objective function and the runtime if the predator is not too active at the beginning of the optimization process and becomes more and more hyperactive with time. This behavior will give a degree of freedom to the GA to explore new solutions without being blocked by the predator. To test it, we modify the predator threshold function for the following one:

$$T_{pred} = \left( 1 + \frac{1}{n_{pred}} \right) \times \text{best objective function value found} \quad (4)$$

To compare the two implementations of the predator,  $P_{pred}$  is set to 0.7. The gain on the computation time between the two predator implementations is near 4%. It is not as much as we

could expect but when analyzing the behavior of those two predators, the first one runs into action before the second one, but the second one is more active at the end of the optimization process. Consequently, the GA has more chances to explore the search space and converge to a better solution with a gain near 8%.

Table D.4 – Predator and Genetic Algorithm – Static vs. dynamic tolerance

( $P_{pred} = 0.7$ )

PGA	Best objective value found	$\Delta_{\text{objective}}$	Runtime (s)	$\Delta_{\text{runtime}}$
Static tolerance (15%)	241.6	-	579.662	-
Dynamic tolerance ( $1/n_{pred}$ )	222.5	-7.92%	556.557	-3.99%

Finally, and to help to measure the gains obtained with the cache memory and predator mechanisms, SGA without memory and without a predator requires a mean of 1100 evaluations ( $N_{ind} = 100$ ,  $G_{gap} = 0.1$ ,  $P_m = 0.6$ ,  $MaxGen = 100$ ) to obtain comparable results of Table D.4 against 180 for the PGA with memory. The best objective value ever found with the PGA is 197.4 and the SGA required 1600 evaluations to find it.

## D.5 Conclusions

In this paper, we have presented two enhancements that can be integrated in a standard genetic algorithm or any hybrid approach. Those enhancements can be useful when the time required for a solution evaluation is long, as is the case in most real life systems. The cache memory mechanism can be used to avoid re-testing a solution generated in past iterations or to constrain the GA to generate singular solutions over iterations, enhancing the search space exploration. The second enhancement consists of a predator implementation in the evaluation process to relax the evaluation of a poor quality chromosome. The combination of those two mechanisms, applied in

a demonstrative example based on a real-life scenario, provides a gain on the best objective function value found by the GA (-19%) and the time consumption (-18%).

It is important to highlight that these two proposed enhancements can be integrated with other heuristics using a long evaluation process such as simulated annealing, tabu search or stochastic gradient. Future works also need to be conducted to better study the impact of the predator mechanisms on the GA performance for different predation thresholds and probability functions. At the same time, the development of new mechanisms to reduce the evaluation time consumption of the GA represents a major field of research for achieving on-line simulation of real and complex manufacturing systems.

## D.6 References

- Azadivar, F., & Lee, Y.-H. (1988). Optimization of discrete variable stochastic systems by computer simulation. *Mathematics and Computers in Simulation*, 30(4), 331-345.
- Azadivar, F., & Tompkins, G. (1999). Simulation optimization with qualitative variables and structural model changes: A genetic algorithm approach. *European Journal of Operational Research*, 113(1), 169-182.
- Baker, J. E. (1985). Adaptive selection methods for genetic algorithms. *Proceedings of the 1st International Conference on Genetic Algorithms* (pp. 101-111).
- Carson, Y., & Maria, A. (1997). Simulation optimization: Methods and applications. *Proceedings of the 1997 Winter Simulation Conference, Atlanta, GA* (pp. 118-126)IEEE Piscataway NJ USA.
- Chaudhry, S. S., & Luo, W. (2005). Application of genetic algorithms in production and operations management: A review. *International Journal of Production Research*, 43(19), 4083-4101.
- Cheng, S.-C., Shiau, D.-F., Huang, Y.-M., & Lin, Y.-T. (2009). Dynamic hard-real-time scheduling using genetic algorithm for multiprocessor task with resource and timing constraints. *Expert Systems with Applications*, 36(1), 852-860.

- Deep, K., & Thakur, M. (2007). A new mutation operator for real coded genetic algorithms. *Applied Mathematics and Computation*, 193(1), 211-230.
- Defersha, F. M., & Chen, M. (2008). A parallel genetic algorithm for dynamic cell formation in cellular manufacturing systems. *International Journal of Production Research*, 46(22), 6389-6413.
- Farhat, F., Nakamura, S., & Takahashi, K. (2009). Application of genetic algorithm to optimization of buckling restrained braces for seismic upgrading of existing structures. *Computers and Structures*, 87(1-2), 110-119.
- Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Boston, MA, USA: Addison-Wesley.
- Gu, J., Gu, X., & Jiao, B. (2008). A Coevolutionary Genetic Based Scheduling Algorithm for stochastic flexible scheduling problem. *7th World Congress on Intelligent Control and Automation, WCICA'08, Chongqing, China* (pp. 4154-4159) Institute of Electrical and Electronics Engineers Inc. Piscataway NJ 08855-1331 United States.
- Holland, J. H. (1975). *Adaptation in Natural and Artificial Systems*: The MIT Press.
- Isgro, F., & Tegolo, D. (2008). A distributed genetic algorithm for restoration of vertical line scratches. *Parallel Computing*, 34(12), 727-734.
- Lee, Y.-H., & Iwata, K. (1991). Part ordering through simulation-optimization in an FMS. *International Journal of Production Research*, 29(7), 1309-1323.
- Mohagheghi, M., & Shayegan, J. (2009). Thermodynamic optimization of design variables and heat exchangers layout in HRSGs for CCGT, using genetic algorithm. *Applied Thermal Engineering*, 29(2-3), 290-299.
- Ogbu, F. A., & Smith, D. K. (1990). Application of the simulated annealing algorithm to the solution of the  $n/m/C_{\max}$  flowshop problem. *Computers & Operations Research*, 17(3), 243-253.



Potter, M. A., & De Jong, K. A. (1994). Cooperative coevolutionary approach to function optimization. *Lecture Notes in Computer Science*(866), 249.

Rossi, A., & Boschi, E. (2009). A hybrid heuristic to solve the parallel machines job-shop scheduling problem. *Advances in Engineering Software*, 40(2), 118-127.

Ruiz, R., Carlos Garcia-Diaz, J., & Maroto, C. (2007). Considering scheduling and preventive maintenance in the flowshop sequencing problem. *Computers and Operations Research*, 34(11), 3314-3330.

Sastry, K., Goldberg, D., & Pelikan, M. (2004). *Efficiency enhancement of probabilistic model building genetic algorithm*. Urbana Champaign, Urbana, IL: Illinois Genetic Algorithms Laboratory , Univeristy of Illinois.

Sastry, K., & Goldberg, D. E. (2007). Let's get ready to rumble redux: Crossover versus mutation head to head on exponentially scaled problems. *9th Annual Genetic and Evolutionary Computation Conference, GECCO 2007, London, United Kingdom* (pp. 1380-1387)Association for Computing Machinery New York NY 10036-5701 United States.

Sastry, K., Lima, C. F., & Goldberg, D. E. (2006). Evaluation relaxation using substructural information and linear estimation. *8th Annual Genetic and Evolutionary Computation Conference 2006, Seattle, WA, United States* (Vol. 1, pp. 419-426)Association for Computing Machinery New York NY 10036-5701 United States.

Taillard, E. (1993). Benchmarks for basic scheduling problems. *European Journal of Operational Research*, 64(2), 278-285.

Tavakkoli-Moghaddam, R., Safaei, N., & Sassani, F. (2009). A memetic algorithm for the flexible flow line scheduling problem with processor blocking. *Computers and Operations Research*, 36(2), 402-414.

Tsai, J.-T., Liu, T.-K., Ho, W.-H., & Chou, J.-H. (2008). An improved genetic algorithm for job-shop scheduling problems using Taguchi-based crossover. *International Journal of Advanced Manufacturing Technology*, 38(9-10), 987-994.

- Whittaker, G., Confesor Jr, R., Griffith, S. M., Fare, R., Grosskopf, S., Steiner, J. J., et al. (2009). A hybrid genetic algorithm for multiobjective problems with activity analysis-based local search. *European Journal of Operational Research*, 193(1), 195-203.
- Yaurima, V., Burtseva, L., & Tchernykh, A. (2008). Hybrid flowshop with unrelated machines, sequence dependent setup time and availability constraints: An enhanced crossover operator for a genetic algorithm. *7th International Conference on Parallel Processing and Applied Mathematics, PPAM 2007, Gdansk, Poland* (Vol. 4967 LNCS, pp. 608-617)Springer Verlag Heidelberg D-69121 Germany.
- Zobolas, G. I., Tarantilis, C. D., & Ioannou, G. (2008). Extending capacity planning by positive lead times and optional overtime, earliness and tardiness for effective master production scheduling. *International Journal of Production Research*, 46(12), 3359-3386.
- Zobolas, G. I., Tarantilis, C. D., & Ioannou, G. (2009). Minimizing makespan in permutation flow shop scheduling problems using a hybrid metaheuristic algorithm. *Computers and Operations Research*, 36(4), 1249-1267.